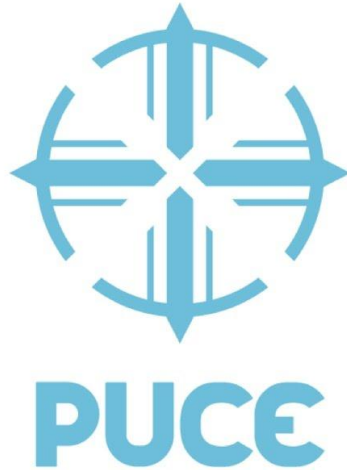


**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR  
FACULTAD DE INGENIERÍA  
ESCUELA DE SISTEMAS**



**DISERTACIÓN PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN  
SISTEMAS Y COMPUTACIÓN**

**“Análisis benchmark de base de datos en memoria,  
base de datos en disco y base de datos mixta”.**

**AUTORES:**

**Derix Paúl Gómez González**

**Sofía Carolina Idrobo Miño**

**QUITO, OCTUBRE 2017**

## Contenido de Disertación

1	Capítulo: Marco Teórico .....	9
1.1	Introducción. ....	9
1.2	Conceptos generales. ....	10
1.2.1	Memorias.....	10
1.2.2	Tipo de Memorias .....	11
1.2.2.1	Memoria principal.....	11
1.2.2.2	Memoria Secundaria.....	12
1.2.3	Gestión de Memorias. ....	12
1.2.4	Gestión de memoria de los Procesos.....	13
1.2.5	Gestión de memoria en Sistemas Operativos.....	13
1.2.5.1	Protección .....	14
1.2.5.2	Memoria compartida .....	14
1.2.5.3	Organización lógica .....	14
1.2.5.4	Organización física .....	15
1.2.5.5	Reubicación .....	15
1.2.6	Bases de datos. ....	15
1.2.7	Gestor de Bases de datos. ....	16
1.2.8	Clasificación de Bases de datos.....	16
1.2.8.1	Según la variabilidad. ....	16
1.2.8.2	Según su contenido. ....	17
1.2.9	Modelos de Bases de datos.....	17
1.2.9.1	Generalidades de las bases de datos relacionales. ....	18
1.2.10	Generalidades de las bases de datos en disco. ....	19
1.2.10.1	Procesador de consultas. ....	21
1.2.10.1.1	Intérprete del Data Definition Language (DDL).....	21
1.2.10.1.2	Compilador del Data Manipulation Language (DML). ....	21
1.2.10.1.3	Motor de evaluación de consultas.....	21
1.2.10.2	Gestor de almacenamiento. ....	21
1.2.10.2.1	Gestor de Archivos. ....	22
1.2.10.2.2	Gestor de Autorización e integridad.....	22
1.2.10.2.3	Gestor de Memoria Intermedia.....	22
1.2.10.2.4	Gestor de Transacciones. ....	22
1.2.11	Generalidades de las bases de datos en memoria principal. ....	22
1.2.12	Generalidades de las bases de datos mixtas.....	24
1.2.13	Definición de análisis benchmark.....	26
2	Capítulo: Actualidad de las bases de datos.....	28
2.1	Software propietario. ....	28
2.2	Software libre.....	28
2.3	Tipos de sistemas de gestión de bases de datos .....	29
2.3.1	Sistemas de gestión de bases de datos propietario.....	29
2.3.2	Sistemas de gestión de bases de datos libres. ....	30

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

2.1	Sistemas de bases de datos en disco en la actualidad. ....	30
2.1.1	PostgreSQL .....	30
2.1.2	MySQL.....	33
2.2	Sistemas de bases de datos en memoria en la actualidad. ....	35
2.2.1	SAP Hana .....	35
2.2.2	Oracle TimesTen In-Memory Database .....	37
2.3	Sistemas de bases de datos mixta en la actualidad. ....	39
2.3.1	Oracle Database 12c con Oracle Database In-Memory .....	39
2.3.2	Microsoft SQL Server 2014 y 2016 .....	40
2.4	Cuadro Resumen Gestores de Bases de datos.....	42
3	Capítulo: Selección de indicadores y ambiente de evaluación.....	43
3.1	Características principales. ....	43
3.2	Selección de sistemas de base de datos.....	44
3.2.1	Selección de Microsoft SQL Server 2016. ....	44
3.2.2	Selección de Oracle Database 12c (Release 2). ....	45
3.2.3	Selección de Oracle TimesTen.....	46
3.3	Materiales y condiciones para la realización de pruebas. ....	47
3.3.1	Diseño de modelo de base de datos. ....	47
3.3.1.1	Modelo Relacional.....	48
3.3.1.2	Modelo Físico .....	48
3.3.2	Software de ayuda.....	48
3.3.2.1	RedGate (SQL Data Generator) .....	48
3.4	Selección de indicadores para la realización del análisis benchmark.....	49
3.4.1	Coincidencias exactas (Q1) (Q2) .....	49
3.4.2	Acceso ordenado (Q3) (Q4) .....	49
3.4.3	Casting (Q5) .....	49
3.4.4	Expresiones regulares (Q6) (Q7) .....	49
3.4.5	Encontrar referencias (Q8) (Q9).....	49
3.4.6	Select Anidado (Q10) .....	50
3.4.7	Joins (Q11) (Q12) (Q13).....	50
3.4.8	Búsqueda por índice (Q14) .....	50
3.4.9	Agregación (Q15) (Q16) .....	50
3.4.10	Inserción (Q17).....	50
3.4.11	Actualización (Q18) (Q19) .....	50
3.4.12	Eliminación (Q20) (Q21) .....	50
3.5	Elaboración de Matrices para el análisis benchmark.....	51
4	Capítulo: Evaluación y Análisis benchmark.....	55
4.1	Pruebas de rendimiento SQL Server 2016.....	55
4.1.1	SQL Server 2016 en disco – Evaluación 1. ....	55
4.1.2	SQL Server 2016 mixta – Evaluación 2. ....	56
4.1.3	SQL Server 2016 en memoria – Evaluación 3.....	56
4.1.4	SQL Server 2016 segunda ejecución sin liberar memoria – Evaluación 4. ....	57

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

4.1.5	SQL Server 2016 promedios – Evaluación 5. ....	57
4.2	Pruebas de rendimiento Database Oracle 12c. ....	58
4.2.1	Database Oracle 12c en disco – Evaluación 1.....	58
4.2.2	Database Oracle 12c mixta – Evaluación 2.....	59
4.2.3	Database Oracle 12c en memoria – Evaluación 3. ....	59
4.2.4	Database Oracle 12c segunda ejecución sin liberar memoria – Evaluación 4. ....	60
4.2.5	Database Oracle 12c promedios – Evaluación 5.....	60
4.3	Pruebas de rendimiento Oracle TimesTen. ....	61
4.3.1	Oracle TimesTen en memoria – Evaluación 3. ....	61
4.3.2	Oracle TimesTen segunda ejecución sin liberar memoria – Evaluación 4. ....	62
4.4	Resultados Finales. ....	62
4.4.1	SQL Server 2016. ....	62
4.4.1.1	Promedios Finales. ....	62
4.4.1.2	Matriz de Calificación. ....	65
4.4.1.3	Promedios Finales Segunda ejecución. ....	66
4.4.1.4	Contraste Primeras ejecuciones vs Segundas ejecuciones sin liberar memoria. 68	
4.4.2	Oracle 12c.....	73
4.4.2.1	Promedios Finales. ....	73
4.4.2.2	Matriz de Calificación. ....	75
4.4.2.3	Promedios Finales Segunda ejecución. ....	76
4.4.2.4	Contraste Primeras ejecuciones vs Segundas ejecuciones sin liberar memoria. 78	
4.4.3	Oracle TimesTen. ....	83
4.4.3.1	Promedios Finales. ....	83
4.4.4	Contraste Bases de Datos.....	83
4.4.4.1	Bases de Datos en Disco.....	83
4.4.4.2	Bases de Datos Mixtas.....	85
4.4.4.3	Bases de Datos en Memoria.....	86
5	Capítulo: Conclusiones, Recomendaciones.....	89
5.1	Conclusiones del análisis benchmark.....	89
5.1.1	Conclusiones SQL Server 2016. ....	89
5.1.2	Conclusiones Database Oracle 12c.....	90
5.1.3	Conclusiones TimesTen. ....	91
5.1.4	Conclusiones Contraste Bases de Datos en Memoria.....	91
5.2	Conclusiones de la Disertación. ....	92
5.3	Recomendaciones.....	93
6	Anexos y Bibliografía.....	95
6.1	Definición de Queries.....	95
6.2	Queries SQL Server 2016.....	96
6.3	Queries Oracle 12c.....	99
6.4	Queries Times Ten.....	102
6.5	Bibliografía.....	105

## Índice de Gráficos

Gráfico 1: Funcionamiento del Computador. ....	11
Gráfico 2: Tipos de Memorias. ....	12
Gráfico 3: Dominios para la relación empleados dentro del modelo relacional. ....	20
Gráfico 4: Comunicación de Memorias en Bases de Datos en Memoria Principal. ....	24
Gráfico 5: Comunicación de Memorias en Bases de Datos Mixtas. ....	26
Gráfico 6: Componentes más importantes de PostgreSQL. ....	32
Gráfico 7: Arquitectura MySQL Server. ....	34
Gráfico 8: Arquitectura SAP HANA. ....	36
Gráfico 9: Componentes de Oracle TimesTen In-Memory Database. ....	38
Gráfico 10: Rendimiento optimizado de Transacciones y Queries. ....	40
Gráfico 11: Estructura de funcionamientos SQL Server 2014. ....	40
Gráfico 12: Estructura de funcionamientos SQL Server 2016. ....	41
Gráfico 13: Forrester Wave: In-Memory Databases, 2017. ....	47
Gráfico 14: Modelo Relacional de la Base de datos de Prueba. ....	48
Gráfico 15: Modelo Físico de la Base de datos de Prueba. ....	48
Gráfico 16: Grafico E1 - En Disco vs Mixta vs En Memoria SQL Server 2016. ....	63
Gráfico 17: Grafico E2 - En Disco vs En Memoria SQL Server 2016. ....	64
Gráfico 18: Grafico E3 - En Disco vs En Memoria SQL Server 2016. ....	64
Gráfico 19: Grafico E1 – En Disco vs Mixta vs En Memoria SQL Server 2016 sin liberar memoria. ....	66
Gráfico 20: Grafico E2 - En Disco vs En Memoria SQL Server 2016 sin liberar memoria. ....	67
Gráfico 21: Grafico E3 – En Disco vs En Memoria SQL Server 2016 sin liberar memoria. ....	68
Gráfico 22: Grafico Primera ejecución en Disco vs Segunda ejecución en Disco sin liberar memoria SQL Server 2016. ....	69
Gráfico 23: Grafico Primera ejecución Mixta vs Segunda ejecución Mixta sin liberar memoria SQL Server 2016. ....	70
Gráfico 24: Grafico Primera ejecución en Memoria vs Segunda ejecución en Memoria SQL Server 2016. ....	71
Gráfico 25: Grafico Primera ejecución en Memoria vs Segunda ejecución en Disco SQL Server 2016. ....	72
Gráfico 26: Grafico E1 - En Disco vs Mixta vs En Memoria Database Oracle 12c. ....	73
Gráfico 27: Grafico E2 - En Disco vs En Memoria Database Oracle 12c. ....	74
Gráfico 28: Grafico E3 - En Disco vs En Memoria Database Oracle 12c. ....	75
Gráfico 29: Grafico E1 - En Disco vs Mixta vs En Memoria Database Oracle 12c sin liberar memoria. ....	77
Gráfico 30: Grafico E2 - En Disco vs En Memoria Database Oracle 12c sin liberar memoria. ..	77
Gráfico 31: Grafico E3 - En Disco vs En Memoria Database Oracle 12c. ....	78

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

Gráfico 32: Grafico Primera ejecución en Disco vs Segunda ejecución en Disco sin liberar memoria Database Oracle 12c.....	79
Gráfico 33: Grafico Primera ejecución Mixta vs Segunda ejecución Mixta sin liberar memoria Database Oracle 12c.....	80
Gráfico 34: Grafico Primera ejecución en Memoria vs Segunda ejecución en Memoria Database Oracle 12c. ....	81
Gráfico 35: Grafico Primera ejecución en Memoria vs Segunda ejecución en Disco Database Oracle 12c. ....	82
Gráfico 36: Grafico Contraste de Bases de Datos en Disco - Database Oracle 12c vs SQL Server 2016. ....	85
Gráfico 37: Grafico Contraste de Bases de Datos Mixtas - Database Oracle 12c vs SQL Server 2016.....	86
Gráfico 38: Grafico Contraste de Bases de Datos en Memoria- Database Oracle 12c vs SQL Server 2016 vs TimesTen. ....	88

## Índice de Tablas

Tabla 1: Relación empleados dentro del modelo relacional. ....	19
Tabla 2: Dominios para la relación empleados dentro del modelo relacional. ....	19
Tabla 3: Cuadro Resumen Gestores de Bases de datos. ....	42
Tabla 4: Versiones de Windows 10 por opción de mantenimiento. ....	43
Tabla 5: Versión 1703 (OS build 15063). ....	44
Tabla 6: DB-Engines Ranking. ....	47
Tabla 7: Modelo Matriz para la comparativa de datos en Disco. ....	51
Tabla 8: Modelo Matriz para la comparativa de datos Mixta. ....	52
Tabla 9: Modelo Matriz para la comparativa de datos en Memoria. ....	52
Tabla 10: Modelo Matriz de calificación para la comparativa en cada Base de Datos. ....	53
Tabla 11: Modelo Matriz de promedios para la comparativa en cada Base de Datos. ....	53
Tabla 12: Modelo Matriz para la comparativa sin liberar memoria en cada Base de Datos. ....	54
Tabla 13: Matriz de toma de datos SQL Server 2016 en disco. ....	55
Tabla 14: Matriz de toma de datos SQL Server 2016 mixta. ....	56
Tabla 15: Matriz de toma de datos SQL Server 2016 en memoria. ....	56
Tabla 16: Matriz de toma de datos SQL Server 2016 sin liberar memoria. ....	57
Tabla 17: Matriz de promedios SQL Server 2016. ....	57
Tabla 18: Matriz de toma de datos Database Oracle 12c en disco. ....	58
Tabla 19: Matriz de toma de datos Database Oracle 12c mixta. ....	59
Tabla 20: Matriz de toma de datos Database Oracle 12c en memoria. ....	59
Tabla 21: Matriz de toma de datos Database Oracle 12c sin liberar memoria. ....	60
Tabla 22: Matriz de promedios Database Oracle 12c. ....	60
Tabla 23: Matriz de toma de datos TimesTen en memoria. ....	61
Tabla 24: Matriz de toma de datos TimesTen sin liberar memoria. ....	62
Tabla 25: Extracto 1 En Disco vs Mixta vs En Memoria - Matriz de promedios SQL Server 2016. ....	63
Tabla 26: Extracto 2 En Disco vs En Memoria - Matriz de promedios SQL Server 2016. ....	63
Tabla 27: Extracto 3 En Disco vs En Memoria - Matriz de promedios SQL Server 2016. ....	64
Tabla 28: Matriz de calificación SQL Server 2016. ....	65
Tabla 29: Extracto 1 En Disco vs Mixta vs En Memoria - Matriz de toma de datos SQL Server 2016 sin liberar memoria. ....	66
Tabla 30: Extracto 2 En Disco vs En Memoria - Matriz de promedios SQL Server 2016 sin liberar memoria. ....	67
Tabla 31: Extracto 3 En Disco vs En Memoria - Matriz de promedios SQL Server 2016 sin liberar memoria. ....	67
Tabla 32: Matriz Primera ejecución en Disco vs Segunda ejecución en Disco sin liberar memoria SQL Server 2016. ....	69
Tabla 33: Matriz Primera ejecución Mixta vs Segunda ejecución Mixta sin liberar memoria SQL Server 2016. ....	70

Tabla 34: Matriz Primera ejecución en Memoria vs Segunda ejecución en Memoria SQL Server 2016.....	71
Tabla 35: Matriz Primera ejecución en Memoria vs Segunda ejecución en Disco SQL Server 2016.....	72
Tabla 36: Extracto 1 En Disco vs Mixta vs En Memoria - Matriz de promedios Database Oracle 12c.....	73
Tabla 37: Extracto 2 En Disco vs En Memoria - Matriz de promedios Database Oracle 12c. ...	74
Tabla 38: Extracto 3 En Disco vs En Memoria - Matriz de promedios Database Oracle 12c. ...	74
Tabla 39: Matriz de calificación Database Oracle 12c.....	76
Tabla 40: Extracto 1 En Disco vs Mixta vs En Memoria - Matriz de toma de datos Database Oracle 12c sin liberar memoria. ....	76
Tabla 41: Extracto 2 En Disco vs En Memoria - Matriz de promedios Database Oracle 12c sin liberar memoria.....	77
Tabla 42: Extracto 3 En Disco vs En Memoria - Matriz de promedios Database Oracle 12c. ...	78
Tabla 43: Matriz Primera ejecución en Disco vs Segunda ejecución en Disco sin liberar memoria Database Oracle 12c.....	79
Tabla 44: Matriz Primera ejecución Mixta vs Segunda ejecución Mixta sin liberar memoria Database Oracle 12c.....	80
Tabla 45: Matriz Primera ejecución en Memoria vs Segunda ejecución en Memoria Database Oracle 12c. ....	81
Tabla 46: Matriz Primera ejecución en Memoria vs Segunda ejecución en Disco SQL Database Oracle 12c. ....	82
Tabla 47: Matriz de promedios finales TimesTen. ....	83
Tabla 48: Matriz de Contraste de Bases de Datos en Disco - Database Oracle 12c vs SQL Server 2016. ....	84
Tabla 49: Matriz de Calificación de Contraste de Bases de Datos en Disco - Database Oracle 12c vs SQL Server 2016. ....	84
Tabla 50: Matriz de Contraste de Bases de Datos Mixtas - Database Oracle 12c vs SQL Server 2016.....	85
Tabla 51: Matriz de Calificación de Contraste de Bases de Datos Mixtas - Database Oracle 12c vs SQL Server 2016. ....	86
Tabla 52: Matriz de Contraste de Bases de Datos en Memoria - Database Oracle 12c vs SQL Server 2016 vs TimesTen. ....	87
Tabla 53: Matriz de Calificación de Contraste de Bases de Datos en Memoria -Database Oracle 12c vs SQL Server 2016 vs TimesTen. ....	87

## **1 Capítulo: Marco Teórico**

### **1.1 Introducción.**

En el siguiente trabajo se busca realizar un análisis que muestre las diferencias en cuanto a tiempos de respuesta arrojados por los gestores de bases de datos más influyentes en la actualidad, mediante pruebas efectuadas con el fin de realizar comparaciones en tres formas de manejo de bases de datos; en Disco, Mixta y en Memoria; para que se pueda obtener información más contrastable en el análisis.

Esta comparativa busca determinar la factibilidad de la implementación de gestores de bases de datos mixtas o en memoria, tomando en cuenta que su implementación en muchos casos requiere una inversión monetaria y de tiempo. Además, la comparativa trata de visualizar los casos en los que mejor se desempeña cada gestor con su forma de manejar de datos.

El presente trabajo consta de seis capítulos dedicados al cumplimiento de esta comparativa. En el capítulo inicial se encuentran los conceptos generales para entender de una mejor forma la comparativa, posteriormente el siguiente capítulo indica la actualidad de los gestores de bases de datos, que contiene información importante acerca de la variedad de gestores en el mercado, qué tan usados son algunos gestores, qué beneficios prestan y cuáles son los considerados más recomendables. Seguido tenemos el capítulo de selección de indicadores y ambiente de pruebas, en este capítulo se indica la preparación del ambiente para realizar la comparativa, qué indicadores se evaluarán y cómo se efectuarán las pruebas. El capítulo cuatro evaluación y análisis benchmark contiene todo lo que se refiere a las pruebas, tomas de datos y análisis. El capítulo cinco interpreta todos los datos obtenidos en el capítulo previo, mostrando conclusiones del análisis y conclusiones generales, así también sus recomendaciones. Y por último el capítulo final abarca los anexos y bibliografía.

## **1.2 Conceptos generales.**

### **1.2.1 Memorias**

La memoria dentro del campo de la informática se refiere al espacio que disponemos para almacenar de forma eficiente programas y datos, se debe tener muy en cuenta que existen registros de datos para las instrucciones en la CPU, pero este espacio es muy limitado y hace que se vuelva indispensable para el funcionamiento del computador el manejo de los diferentes tipos de memorias. (Canto, n.d.).

La forma en que la memoria codifica la información es por medio de un alfabeto binario conformado por bits, comúnmente llamado octeto o byte, esta es la medida que se maneja para dar valor a la capacidad que tiene de almacenamiento dicha memoria.

Para brindar un mejor entendimiento de la interacción entre las memorias y los diferentes componentes del computador, se muestra en la figura 1, donde el procesador se comunica con la memoria principal para obtener las instrucciones y datos, las cuales han sido recibidos por los periféricos de entrada y serán presentados por los de salida después de realizar su procesamiento. El control en el microprocesador permite administrar las operaciones de datos, la memoria, la entrada y la salida.

La memoria principal además obtiene datos de la memoria secundaria cuando el procesador requiere información almacenada en dicha memoria. Este proceso se explicará de mejor manera más adelante.

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

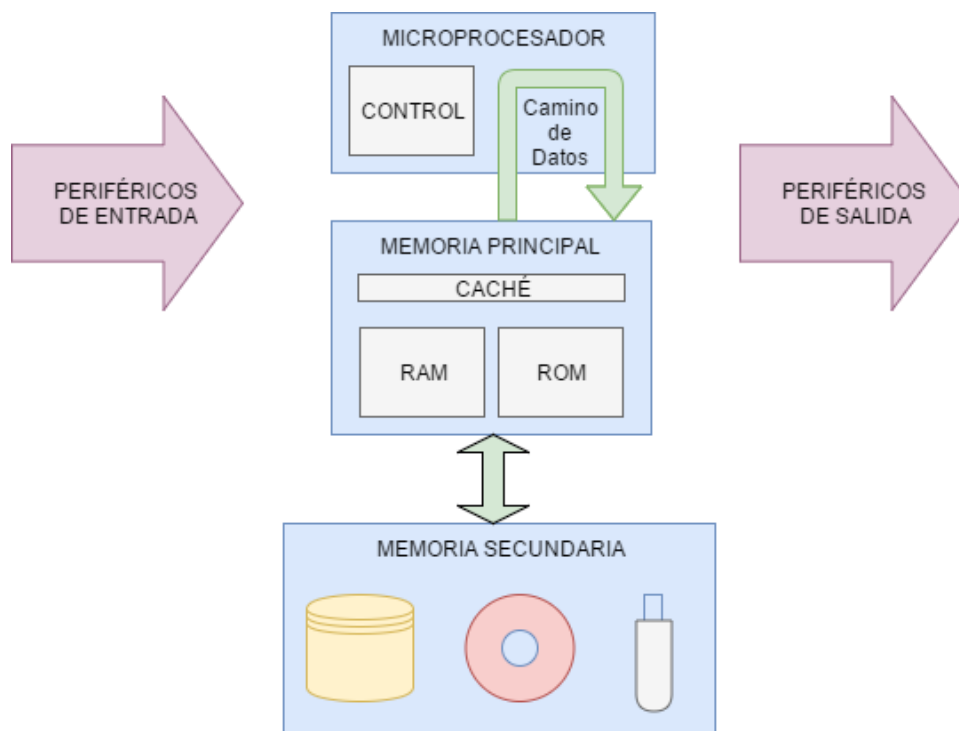


Gráfico 1: Funcionamiento del Computador.

(Patterson & Hennessy, 2011; Puertas Serrano & Delibes, 2007).

## 1.2.2 Tipo de Memorias

### 1.2.2.1 Memoria principal

Esta memoria es la más veloz ya que tiene como función el almacenar los datos que se están ejecutando en el ordenador, la memoria se encuentra dividida en posiciones o también llamadas celdas conocidas como palabras de memoria, el tamaño de esta memoria es normalmente definido por múltiplos del byte: 8, 16, 32 etc.

Dentro de la memoria principal encontramos las memorias de lectura y escritura, que son denominadas como RAM estas son de característica volátil, además encontramos las memorias de solo escritura ROM de característica no volátil. Es importante definir que la memoria volátil pierde la información al cortar la corriente y por el contrario la memoria no volátil no la pierde.

### 1.2.2.2 Memoria Secundaria

A diferencia de la limitada capacidad de almacenamiento en la memoria principal, la memoria secundaria posee mayores cantidades de almacenaje, al tener más espacio la transferencia de datos se vuelve más lenta “del orden de mil veces más lentas, pero más capaces” (Tellez, n.d.).

Estas memorias adoptaron el concepto de fichero o archivo, esto hace una referencia a una cantidad de datos relacionados entre sí.

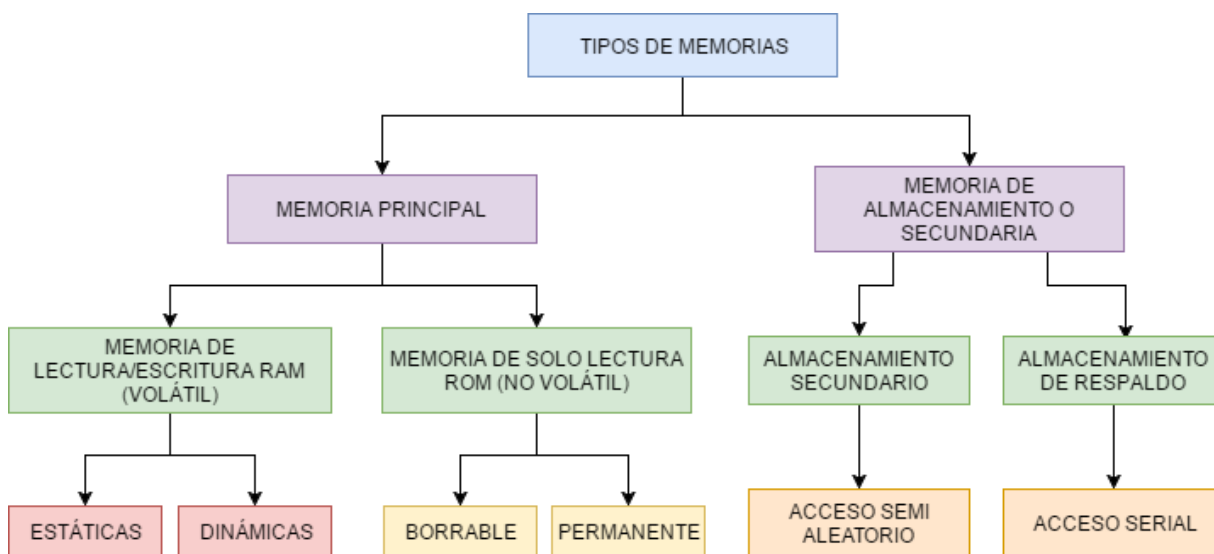


Gráfico 2: Tipos de Memorias.

(Canto, n.d.)

### 1.2.3 Gestión de Memorias.

La gestión de memoria en la informática es muy importante ya que es uno de los principales recursos con los que cuentan la mayoría de dispositivos tecnológicos, en la antigüedad se tenía mucha limitación en cuanto a este recurso, pero ya que actualmente se cuenta con una mayor cantidad de memoria a un menor costo se puede llegar a descuidar su gestión, el tener más memoria no exime que se debe ser cuidadoso en su gestión.

Visto de una forma sencilla la gestión de memoria se centra en la asignación de memoria en donde sea requerida. (Romero, 2009)

#### **1.2.4 Gestión de memoria de los Procesos.**

Para aumentar las prestaciones de una computadora es necesaria además de la compartición del procesador, la compartición de la memoria, dado que actualmente los usuarios requieren la respuesta de varios procesos a la vez y las computadoras están enfocadas al multiproceso.

La memoria está compuesta de una matriz de palabras o bytes con su respectiva dirección de memoria. Los procesos dan ordenes que son ejecutadas por la CPU (Central Processing Unit), las cuales extraen una instrucción de memoria y pueden provocar operaciones de almacenamiento o carga a direcciones de memoria. De esta forma, los procesos en ejecución generan un flujo de direcciones de memoria en orden secuencial.

Sin embargo, para que la CPU pueda ejecutar un programa, necesita que las instrucciones y datos se encuentren en memoria, debido a que la memoria principal y los registros son los únicos dispositivos de almacenamiento a los cuales esta unidad puede acceder directamente. Por esta razón cada programa debe estar asignado a ciertas direcciones de memoria para ser ejecutado, una vez que un programa finaliza, libera las direcciones para que otro programa pueda acceder a estas.

Para brindar una mejor respuesta a los usuarios, las computadoras mantienen distintos programas en memoria, para lo cual se requiere mecanismos de gestión de memoria que permitan administrar correctamente cada una de las direcciones a los distintos programas. (A. Silberschatz, Galvin, & Gagne, 2006)

#### **1.2.5 Gestión de memoria en Sistemas Operativos.**

La memoria principal tiene un costo elevado en el mercado, comparado con el resto de recursos que conforman los dispositivos, por lo tanto, los sistemas operativos deben estar diseñados con el objetivo de optimizar el uso de esta memoria.

Para comenzar con la gestión de memoria en los sistemas operativos, es importante indicar la diferencia entre una dirección lógica y una física. La primera es aquella dirección generada por la CPU y la segunda por la memoria principal.

La tarea de correspondencia entre las direcciones lógicas y físicas la realiza la unidad de gestión de memoria (MMU), la cual es un dispositivo de hardware parte de procesador que además se encarga de convertir las direcciones lógicas en físicas siempre y cuando esta conversión sea posible y tenga los permisos asociados.

Esta unidad se asigna para cada proceso del sistema y cuando existe un error es la parte asociada a la administración de memoria, que es en realidad parte del Kernel del sistema operativo, el encargado de resolver este error.

El Kernel se encarga de mantener un registro de las direcciones de memoria utilizadas y libres, administra el espacio otorgado de memoria principal y el intercambio de datos entre memoria principal y secundaria.(Almería, n.d.)

De esta forma los programas nunca ven las direcciones físicas reales, estos generan direcciones lógicas que son convertidas en físicas para poder utilizarlas.

Además, dentro de los sistemas operativos, se debe asociar a la gestión de la memoria con las siguientes tareas:

#### **1.2.5.1 Protección**

La gestión de memoria debe verificar el permiso otorgado a un proceso para referenciar un espacio de memoria, así se evita que un proceso pueda acceder a espacios que no le han sido asignados o modificar direcciones de otro proceso.

#### **1.2.5.2 Memoria compartida**

La gestión de memoria debe permitir que cada proceso pueda acceder a espacios de memoria y compartir información, así se evita asignar diferentes espacios de memoria con la misma información y permite que los procesos se comuniquen de forma más rápida.

#### **1.2.5.3 Organización lógica**

La gestión de memoria debe mantener la organización por módulos en la cual están escritos los programas, y ejecutar estos módulos por separado con su diferente nivel de protección. (A. Silberschatz et al., 2006)

#### **1.2.5.4 Organización física**

La gestión de memoria debe permitir el traslado de datos entre el almacenamiento primario y el secundario. (A. Silberschatz et al., 2006)

#### **1.2.5.5 Reubicación**

En cada ejecución de un programa, como este va a compartir memoria con otros procesos, dicho programa puede ser asignado a diferentes áreas de memoria. Así mismo, durante su ejecución el programa puede salir de la memoria principal en momentos y regresar a ocupar distintos espacios de memoria. La gestión de memoria debe permitir reubicar las referencias de memoria de cada proceso de manera correcta y continua. (Almería, n.d.)

#### **1.2.6 Bases de datos.**

Visto de una manera general una base de datos es una colección de datos relacionados entre sí, que permiten el almacenamiento de grandes cantidades de información, por ejemplo, una base de datos sería los registros de varias personas y sus atributos como nombres, apellidos y direcciones, etc. Todos estos se almacenan y administran en el disco duro de un computador mediante una aplicación como un ejemplo Microsoft SQL Server. Por esta razón, las bases de datos en la actualidad ocupan un papel principal en la mayoría de áreas donde se maneje tecnología.

Una visión más restringida de este término adopta algunas propiedades implícitas. La primera se refiere a la representación de la información que se encuentra en la base de datos, esta debe corresponder a un aspecto del mundo real y los cambios deben ser actualizados lo más rápido posible. La segunda indica que debe existir un nivel de coherencia en información que se encuentra almacenada, no es correcto decir que una base de datos es una cantidad de datos aleatorios. La tercera propiedad plantea que las bases de

datos deben cumplir un propósito de y además estar dirigida a un grupo de usuarios interesados.

### **1.2.7 Gestor de Bases de datos.**

Los sistemas gestores de base de datos SGBD o por sus siglas en inglés DBMS (Data Base Management System) son un conjunto de programas que permiten a distintos usuarios finales y administradores del sistema construir, manejar y compartir bases de datos.

Para que el SGBD pueda manipular y almacenar una base de datos es importante que también almacene el catálogo de la base de datos, donde se indique y especifique los tipos de datos, la estructura de los datos y las políticas o restricciones de los mismos. De esta forma, el SGBD puede controlar las operaciones que se apliquen a la base de datos, como consultas de datos o actualizaciones, dado que conoce los datos que mantiene almacenados.

Un SGBD usualmente ayuda a la compartición de los datos entre varios usuarios, controlando que la actualización o concurrencia de los datos sea correcta, y a la protección de la base de datos otorgando distintos niveles de acceso a los usuarios. Además, el SGBD permite brindar a los usuarios una visión de los datos sin incluir características de su almacenamiento o implementación. (Elmasri & Navathe, 2014).

### **1.2.8 Clasificación de Bases de datos.**

Las bases de datos se pueden clasificar por el contenido de los datos almacenados, también visto como el propósito de esos datos, y también por la estructura en que esta está organizada.

#### **1.2.8.1 Según la variabilidad.**

Si nos referimos a la variabilidad de los datos que encontramos en la base, podemos diferenciar dos categorías estructurales la una que realiza dicha variación y la otra no, la primera se refiere a las bases de datos dinámicas y la segunda a las bases de datos estáticas.

Las bases de datos dinámicas son aquellas que permiten realizar inserciones, modificaciones y eliminaciones de los datos que se encuentran alojados en la base, esto permite que con el tiempo la información se actualice. Un ejemplo es las bases de datos tipo OLTP (On Line Transaction Processing), que serían bases de datos dinámicas pero orientadas al manejo de transacciones, normalmente usadas en Business Intelligence. (Anguiano Morales, 2014)

Las bases de datos estáticas son aquellas que principalmente se destinan para la lectura de información esto nos indica que no se inserta, no se modifica ni se elimina, solo se consulta la información que se encuentra alojada en la base de datos. Un ejemplo es las bases de datos de tipo OLAP (On Line Analytical Processing) que mantiene la estructura estática, se orienta al procesamiento analítico y se implementa para aumentar el desempeño de las consultas en grandes cantidades de data.

#### **1.2.8.2 Según su contenido.**

Esta clasificación se identifica por el contenido de los datos que están almacenados en la base de datos, por ejemplo, las bases de datos bibliográficas. Actualmente, son muy importantes ya que, al ser consideradas una herramienta para la gestión de la información, se las puede considerar como las fuentes secundarias, ya que nos conducen a la fuente original, incluso pueden contar con un extracto o resumen de la fuente original. De igual forma podríamos mencionar otros enfoques de información en otros campos como los directorios o las bibliotecas.

#### **1.2.9 Modelos de Bases de datos.**

Como se había mencionado anteriormente, los Sistemas Gestores de Bases de Datos, no muestran los datos junto con su estructura de almacenamiento, sino que brindan una visión abstracta de los mismos.

Estos conceptos de cómo está organizada su estructura, cómo se realiza el almacenamiento y recuperación de datos generan el modelo de la base de datos.

Por lo tanto, podemos concluir que un modelo de base de datos representa una descripción de la organización y la gestión de los datos.

Los modelos de base de datos se clasifican por el nivel en que describen la estructura de los datos.

Primero, tenemos el modelo conceptual o de alto nivel, donde los datos están descritos como la mayoría de las personas los percibe en el mundo real, se puede decir que está orientado a usuarios finales. A diferencia del modelo conceptual, el modelo físico o de bajo nivel, está orientado a usuarios especializados dado que brindan información del almacenamiento de los datos en el disco del dispositivo. Finalmente, el modelo lógico, que podríamos decir, combina los dos últimos modelos mencionados, brindando cierta información de su almacenamiento y representaciones de datos orientados a un usuario final, son modelos que se organizan en registros u objetos. De este último, el modelo lógico, podemos mencionar algunos ejemplos como, el modelo relacional, jerárquico, de red u orientado a objetos. (Marqués, 2009)

#### **1.2.9.1 Generalidades de las bases de datos relacionales.**

Las bases de datos relaciones son aquellas que adoptan el modelo relacional para su construcción y manejo de datos.

El modelo relacional de bases de datos fue creado por el informático Edgar Frank Codd en 1970. Este modelo se basa en el concepto matemático de tabla de valores y teoría de conjuntos.

En el modelo relacional, las relaciones son tablas que en cada registro o fila de la misma, se observa un conjunto de valores que se encuentran relacionados. En este modelo cada fila o registro es denominado tupla y cada uno de los valores o columnas son atributos de la relación.

Es importante mencionar que el nombre de las relaciones y atributos son únicos, además, a cada tupla se le puede asignar un valor único en un determinado atributo que la identifique.

Dentro de este modelo, también es necesario mencionar el concepto de dominio. Dominio es aquel conjunto de valores permitidos que se pueden asignar a un atributo o a varios de ellos.

En el siguiente ejemplo se muestra la relación Empleados para mostrar la información de los trabajadores de una organización, con las columnas para los atributos Código (para código de empleado), Nombre (nombre del empleado), Apellido (apellido del empleado), Edad (edad del empleado) y Departamento (departamento asignado del empleado).

Codigo	Nombre	Apellido	Edad	Ddepartamento
333	Luis	Torres	22	SISTEMAS
351	Anabel	Torres	26	CONTABILIDAD
866	Maria	Castilla	25	GERENCIA

Tabla 1: Relación empleados dentro del modelo relacional.

Y como ejemplos de dominio que se pueden aplicar a estos atributos:

Departamento: Conjunto de los nombres de departamentos en la organización.

Edad: Conjunto de edad permitida para los empleados de una empresa, de 18 a 80.

Dominio	Atributo	Detalle	Definición
dom_edad	Edad	Posible edad de empleado.	Número entero entre 18 y 80.
dom_dep	Departamento	Nombre de Departamentos.	Cadena de 50 caracteres.

Tabla 2: Dominios para la relación empleados dentro del modelo relacional.

Como conclusión, se puede entender que una base de datos relacional es un conjunto de tablas donde cada tupla es una instancia determinada por sus atributos interrelacionados y definidos por un dominio. (Elmasri & Navathe, 2014)

### 1.2.10 Generalidades de las bases de datos en disco.

Este tipo de bases de datos nos permiten el almacenamiento de gran cantidad de información siendo una ventaja importante, para tener a nuestra disposición gran cantidad de espacio destinadas al almacenamiento, pero también se pudiera ver como una desventaja ya que a mayor tamaño de la base de datos

se exige mayor espacio de disco y más memoria RAM para manejarla de una forma eficiente, esto conlleva más costos. El manejar grandes cantidades de información en el disco duro hace que las tareas de lectura y escritura se vuelvan un poco más lentas y en ocasiones se generen cuellos de botella ya que la velocidad que brinda un disco duro es la más lenta con respecto a los discos de estado sólido (SSD Solid-State Drive) y mucho más lenta que la que ofrece la memoria principal.

En la siguiente figura, podemos observar la comunicación que existe entre memoria principal y memoria secundaria, mostrando que la información es llamada de la memoria secundaria a la memoria principal dependiendo de las instrucciones enviadas por las diferentes aplicaciones que soliciten la información.

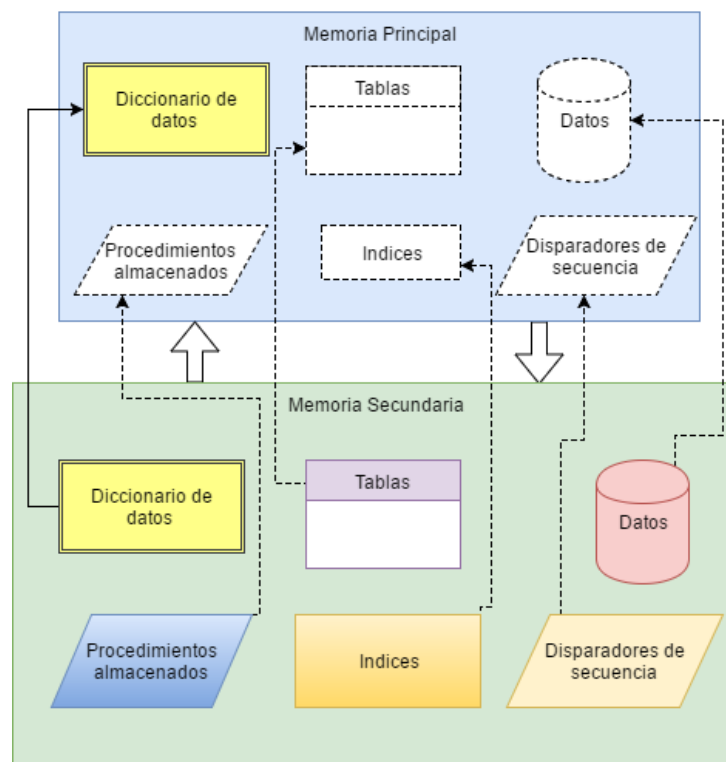


Gráfico 3: Dominios para la relación empleados dentro del modelo relacional.

Para entender cómo funcionan las bases de datos en disco o “tradicionales”, tenemos que describir cómo opera su estructura más común, es importante notar que existen dos componentes principales. Estos son el procesador de consultas y el gestor de almacenamiento. (A. (Bell L. Silberschatz, Korth, & Sudarshan, S. (Instituto Indio de Tecnología, 2002)

### **1.2.10.1 Procesador de consultas.**

El procesador de consultas es aquel que transforma las consultas realizadas por el usuario a un lenguaje de bajo nivel, sus componentes son el motor de evaluación de consultas, el intérprete del DDL “Data Definition Language” y el Compilador del DML “Data Manipulation Language”.

#### **1.2.10.1.1 Intérprete del Data Definition Language (DDL).**

Como su nombre lo indica, es aquel que se encarga de interpretar las instrucciones de tipo DDL (create, alter, drop, truncate) que alteran la estructura de la base de datos. Y posteriormente llevan las definiciones al diccionario de datos.

#### **1.2.10.1.2 Compilador del Data Manipulation Language (DML).**

Este componente se define como el que toma las instrucciones de tipo DML (insert, delete, update) y las convierte en instrucciones que el motor de evaluación de consultas es capaz de entender (lenguaje de bajo nivel). Además, durante la traducción que realiza este compilador, verifica que se genere la instrucción más eficiente mediante una optimización de consultas.

#### **1.2.10.1.3 Motor de evaluación de consultas**

El motor de evaluación de consultas es el encargado de ejecutar las instrucciones que el DML compila a lenguaje de bajo nivel.

### **1.2.10.2 Gestor de almacenamiento.**

Este gestor es un módulo que recibe del motor de evaluación las instrucciones DML ya en bajo nivel, el gestor de almacenamiento suministra la interfaz entre las instrucciones, consultas, programas de aplicación y también tiene como dos importantes responsabilidades que son la interacción con el gestor de ficheros y la actualización, recuperación y almacenamiento eficiente de datos. Sus componentes son:

#### **1.2.10.2.1 Gestor de Archivos.**

Es aquel que se encarga del manejo del espacio con el que cuenta el disco lo que incluye la reserva de espacio y además la muestra de los datos que se encuentran guardados en el disco.

#### **1.2.10.2.2 Gestor de Autorización e integridad.**

Este componente es el encargado de la seguridad, su tarea es realizar una revisión para verificar que se cumplan las limitaciones de integridad, también se encarga de realizar una validación de los usuarios, para comprobar los privilegios que tienen de acceso a la información.

#### **1.2.10.2.3 Gestor de Memoria Intermedia.**

Este componente es el que maneja la información del disco a la memoria principal y es aquel que decide cuando mover dicha información a la memoria cache. El gestor de memoria intermedia hace posible el manejo de información, inclusive de mayor tamaño de almacenamiento, que el disponible en la memoria principal. También es capaz de descartar o sustituir los espacios en la memoria interna o espacio en el buffer cuando no exista espacio o cuando se lo requiera para atender la petición del sistema de base de datos, esto lo hace mediante Less Recently Used (Usado menos recientemente) o Most Recently Used (Usado más recientemente). Estas estrategias reducen los accesos al disco.

#### **1.2.10.2.4 Gestor de Transacciones.**

Este gestor es el que se encarga de que las transacciones se realicen o no se realicen, en otras palabras, si una transacción no es completada no se realice una modificación que provoque inconsistencias en la base de datos. (A. (Bell L. Silberschatz et al., 2002)

### **1.2.11 Generalidades de las bases de datos en memoria principal.**

Mientras las bases de datos tradicionales se almacenan en medios persistentes como discos de almacenamiento, las bases de datos en memoria, como su nombre lo dice, se almacenan en la memoria principal de un ordenador y son gestionadas por un SGBD en memoria.

El principal beneficio de emplear estas bases de datos es el desempeño que brindan a consultas y operaciones. Como los datos están presentes en memoria todo el tiempo, no existe latencia en la lectura de datos, las actualizaciones son casi inmediatas y las relaciones entre datos se manejan a través de punteros directos. Otra razón por la cual mejora el rendimiento es no tener latencia al evitar lecturas y escrituras de disco, cuando el gestor de la base de datos gana el procesador, no tiene que llamar al sistema operativo por que los datos se encuentran listos en memoria, y de esta forma evita abandonar la unidad de procesamiento y pasar a espera.

Al hablar de almacenamiento en la memoria principal, general mente en la memoria RAM, se debe mencionar el riesgo de pérdida de datos al ser una memoria volátil. Si el ordenador pierde energía, esta memoria perderá la información que no se haya guardado. Por esta razón es importante que en las bases de datos en memoria se administre los puntos de guardado, o como se los conoce mejor por su nombre en inglés “save points”, según un tiempo aceptable requerido para mantener el rendimiento necesario reduciendo el riesgo a cierto nivel.

En las bases de datos en memoria los datos están listos para utilizarse y permiten un acceso mediante columnas, filas, índices y permite reasignar punteros. Estos datos se modifican de manera atómica, esto se refiere a que las transacciones solo se realizan si se completan de forma satisfactoria, si una parte de esta falla no se realizaran cambios a los datos.

Algunos sistemas gestores de bases de datos en memoria cuentan con herramientas para aumentar la persistencia o la seguridad de la información en este tipo de bases de datos. Algunas de ellas permiten la escritura de los save points en medios no volátiles que permitan indicar la última transacción y ejecutar las que se hayan perdido en caso de falla. Algunos también, permiten mantener una copia de seguridad mediante replicación. O se puede aplicar el uso de una memoria NVRAM, la cual funciona como una memoria RAM, pero

cuenta con una batería que evita la pérdida de información al detenerse la alimentación de energía. (McObject LLC, 2017; RAIMA INC., 2017)

En la siguiente figura, podemos observar la comunicación que existe entre memoria principal y memoria secundaria en las bases de datos en memoria principal, mostrando que la información es gestionada plenamente en la memoria principal comunicándose con la memoria secundaria en el momento en que se requiera llevar la información de la memoria secundaria a la memoria principal o viceversa en caso de que se requiera almacenar la información en un medio persistente como medida de seguridad.

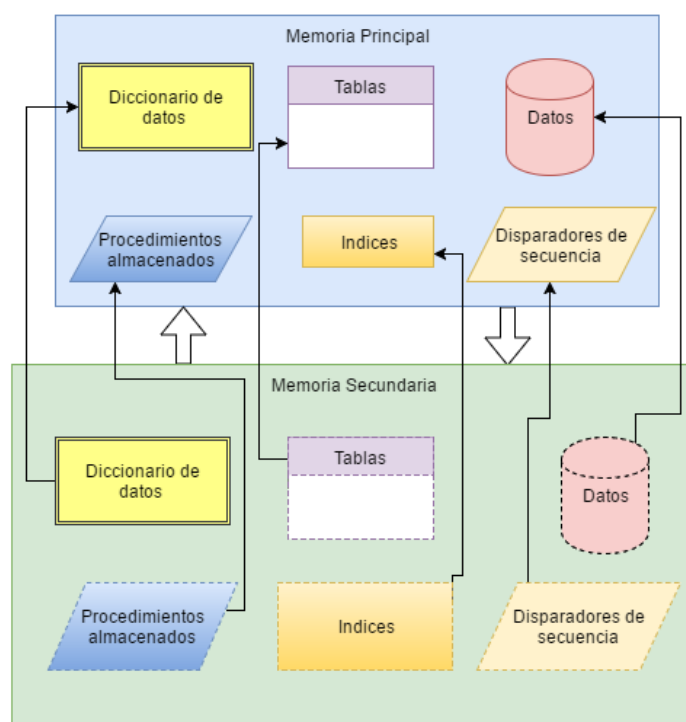


Gráfico 4: Comunicación de Memorias en Bases de Datos en Memoria Principal.

### 1.2.12 Generalidades de las bases de datos mixtas.

Las bases de datos mixtas también llamadas híbridas son capaces de ocupar el almacenamiento en memoria principal y en una unidad de disco. Por esta razón, obtienen un alto rendimiento, así como una mayor persistencia o durabilidad al emplear una memoria no volátil para almacenar los datos. Es importante mencionar, que la mayoría de estas bases de datos han surgido como son bases de datos en disco, que buscando optimizar su procesamiento con una reducción de costos de la memoria RAM, han visto como una oportunidad utilizar algunos objetos en memoria principal.

La principal razón para emplear una base de datos híbrida, es para lograr un equilibrio entre el rendimiento y la durabilidad, la durabilidad en base de datos se entiende como la capacidad de que los datos puedan persistir en el tiempo. Además, se debe mencionar que al utilizar una base de datos en memoria su costo es el más elevado de implementar, debido a que la memoria principal es más costosa en comparación con la memoria de discos de almacenamiento. También, otro factor importante a mencionar es que muchas veces las bases de datos alcanzan tamaños muy grandes y su almacenamiento en memoria principal podría resultar en un alto costo. De esta forma, si se utiliza una base de datos mixta se puede realizar un ahorro en cuanto a la memoria de disco. Y también, se puede manejar mayores cantidades de información aprovechando el tamaño de la memoria en disco.

Finalmente, se puede decir que el rendimiento de la base de datos mixta está en un punto medio entre la base de memoria y la base de datos en disco. Por lo cual se puede emplear para bases OLTP u OLAP. (“What is a Hybrid Database?,” n.d.)

Como ejemplo de este tipo de bases de datos, tenemos a SQL Server 2016 (“SQL Server 2016 | Microsoft,” 2017), la cual es capaz de aprovechar un mejor rendimiento de los datos en memoria, así como también de las características del almacenamiento en disco.

En la siguiente figura, podemos observar la comunicación que existe entre memoria principal y memoria secundaria en las bases de datos mixtas, mostrando que la información es gestionada parcialmente la memoria principal y así también, en la memoria secundaria, comunicándose entre sí cuando sea requerido.

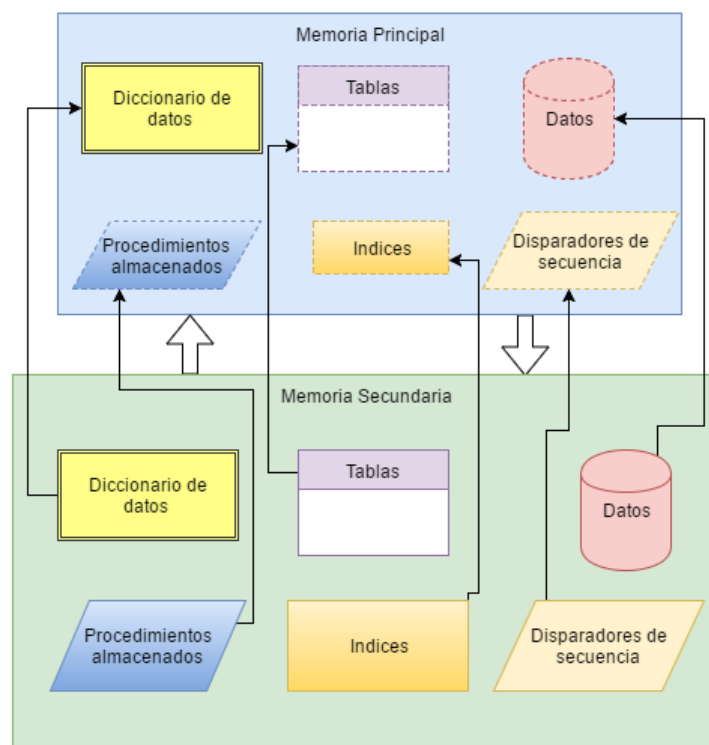


Gráfico 5: Comunicación de Memorias en Bases de Datos Mixtas.

### 1.2.13 Definición de análisis benchmark.

El término benchmark es un anglicismo que se traduce como comparativa en el idioma español, por lo tanto, benchmark se define como la realización de un análisis comparativo de servicios, productos, componentes, aplicaciones, etc. Para la realización de una comparativa se debe definir los criterios de evaluación, así como también el elemento que se evaluará, los elementos con los que se comparará y con esta información se puede obtener resultados que brinden un panorama que permita la elección del elemento que sea considerado el más eficiente en el campo que se lo evaluó.

En el campo de la informática, se han desarrollado diferentes aplicaciones o programas que ayudan, a usuarios comunes o con mayor conocimiento en dicho campo, a realizar benchmark enfocado a medir el rendimiento de un dispositivo, así como sus diferentes componentes. Este programa expone al dispositivo a diversos estímulos o tareas para determinar la productividad o la forma de trabajar bajo las mismas. Además de ayudar a determinar la eficiencia y la integridad de los dispositivos sometidos a la comparativa, permite conocer aspectos técnicos del mismo.

De esta manera, el usuario puede identificar qué componentes se pueden mejorar o la compatibilidad entre los mismos, así como identificar las mejores características técnicas para realizar tareas en específico. (López & Gutiérrez, 2009)

En la actualidad, el benchmark se ha popularizado dentro de los dispositivos móviles gracias a la amplia gama de opciones que el mercado ofrece. Por otro lado, dentro del campo de las bases de datos existen varios sistemas que permiten realizar benchmark, como: Database Benchmark (Borghello, 2016), TPC Benchmarks (“TPC-Homepage V5,” 2017), Xbench, DWEB (Darmont, n.d.), los cuales están orientados a la evaluación de distintos aspectos de las mismas, así como, a diferentes sistemas gestores de bases de datos.

## 2 Capítulo: Actualidad de las bases de datos

### 2.1 Software propietario.

El software propietario actualmente cuenta con muchos términos para llamarlo. Culebro, Gómez y Torres (2006) afirman: “El software no libre también es llamado software propietario, software privativo, software privado o software con propietario.” (p. 3). Cada uno de estos nombres tiene una diferente difusión por el público en internet o por expertos, así como diferentes objeciones a su nombre por el significado que puede brindar el mismo. Sin embargo, estos términos buscan agrupar al software que se encuentra bajo los derechos de autor de una persona física o jurídica, bajo el sistema copyright, limitando a los usuarios sus oportunidades de estudiar el funcionamiento del código o acceder al código fuente, realizar modificaciones al software o mejorarlo y distribuir dichas mejoras, realizar copias, o si mantiene derechos sobre el uso de este. La distribución de este software depende de la empresa desarrolladora, la licencia y el contrato que esta exija. (Culebro Juárez, Gómez Herrera, & Torres Sánchez, 2006)

### 2.2 Software libre.

Richard Stallman en el libro “Software libre para una sociedad libre” indica, que cuando hablamos de software libre este no se refiere a precio, nos debemos centrar en detalles concretos, por ejemplo, las libertades más importantes que debemos cumplir para considerar un software como libre.

Estas libertades son:

- Libertad Cero, el poder que se otorga de ejecutar el programa, sin importar el propósito que se pudiera tener con el mismo.
- Libertad Uno, el poder realizar un estudio o modificación del programa, para que se pueda amoldar mucho mejor a las necesidades que se tengan.
- Libertad Dos, el poder crear copias del software y además el poder distribuirlas. Libertad tres, el poder crear una versión mejorada del

software y así mismo poder construir una comunidad en base a esas mejoras para que todos puedan beneficiarse de las mismas.

Resumiendo, podríamos concluir que el software libre es aquel que puede ser distribuido, modificado, copiado y usado.

Además, es importante mencionar otro concepto concreto que protege al software libre cuando principalmente se da un paso más adelante y se distribuye dicho software.

Copyleft. - Es aquel que impide a aquellos que deseen realizar una redistribución de software, la inclusión de restricciones que atenten con las libertades que fue concebido, contrario a copyright que según Stallman solo adopta la libertad cero. (Stallman, 2004)

### **2.3 Tipos de sistemas de gestión de bases de datos.**

Al momento de clasificar los sistemas gestores de bases de datos, podemos dividirlos en base a muchos criterios, por ejemplo: en cuanto a su modelo de datos, el número de usuarios, siendo monousuario y multiusuario, o incluso según el número de sitios, que serían los distribuidos o centralizados, entre otros. Pero la clasificación por la que los diferenciaremos será referente a si son propietarios o libres.

#### **2.3.1 Sistemas de gestión de bases de datos propietario.**

Muchas compañías de software han desarrollado sistemas gestores de bases de datos liberándolos como software propietario, como ejemplo de estos gestores están: Microsoft SQL Server, Oracle, Paradox, IBM DB2, IBM Informix, etc.

Es decisión de cada usuario el optar por este tipo de sistemas gestores dependiendo de sus características, necesidades de usuario y ventajas que muchos de estos ofrecen, por ejemplo: por lo general se pueden firmar contratos de mantenimiento y soporte, los programas pasan por un riguroso control de calidad, así como actualizaciones de seguridad, paquetes, etc., cuentan con software para aplicaciones específicas, las empresas desarrolladoras ofrecen paquetes empresariales y educativos, entre otras

ventajas. No obstante, estos sistemas se enfrentan a las desventajas del software propietario, donde no se puede acceder al código fuente, es difícil o costoso realizar adaptaciones personalizadas a módulos del programa y la dependencia a una compañía que puede discontinuar el software. (Culebro Juárez et al., 2006)

### **2.3.2 Sistemas de gestión de bases de datos libres.**

Tomando como base el concepto de software libre podemos detallar lo que serían los sistemas de bases de datos libres, entre los sistemas que respetan las libertades para poder catalogarse como libres tenemos: MySQL, SQLite, PostgreSQL, Firebird, Apache Derby, entre otros.

Cuando hablamos de estos gestores podemos decir que los sistemas de gestión de bases de datos libres tienen más ventajas que desventajas, por ejemplo, cuando nos referimos al costo, algo que no define si un gestor es libre o no. Podemos decir que al ser libre es un poco más limitada su venta y normalmente se buscan alternativas de modelos de negocio para conseguir financiamiento, por esto si el gestor tiene costo, este suele ser menor al de los privados. Otro punto importante a tener en cuenta es el soporte, porque normalmente se cree que al ser libre no se obtiene soporte, pero esto es falso, ya que al ser libres se permite un mayor manejo para los usuarios y esto hace que se creen foros o comunidades, donde se encuentre un soporte de gran calidad. Además, el poder modificar estos sistemas los vuelven más resistentes a la obsolescencia y también hace que se ajuste mejor a las necesidades.

Uno de los puntos negativos podría ser la ausencia de una garantía ya que esta es procedente del autor, otro punto negativo sería la dificultad de manejo ya que los sistemas propietarios y sus configuraciones suelen ser más intuitivos que los libres. (Culebro Juárez et al., 2006; González Barahona, Seoane Pascual, & Robles, 2003)

## **2.1 Sistemas de bases de datos en disco en la actualidad.**

### **2.1.1 PostgreSQL**

PostgreSQL es un gestor de bases de datos relacional actualmente muy usado, incluso en su página oficial, catalogan a PostgreSQL como el gestor de bases de datos de código libre más avanzado y potente del mercado. Este gestor es distribuido bajo la licencia BSD, que hace referencia a una licencia de software libre, esta permite el manejo del código fuente en software no libre.

PostgreSQL maneja el modelo cliente/servidor y además no usa multihilos sino más bien para brindar una mejor estabilidad del sistema usa multiprocesos. En cuanto a los límites del gestor se ha estimado que el tamaño máximo de una tabla es de 32 TB, una fila 1.6 TB, el de un campo es 1GB, el número de filas y número de índices por tabla es ilimitado, en cuanto al número de columnas está entre 250 y 1600 dependiendo el tipo de dato y por último el tamaño total de la base de datos dependerá de la capacidad de almacenamiento con la que cuente el ordenador donde funciona.

La última versión estable de PostgreSQL es la 9.6 y brinda como características mejoradas, la ejecución en paralelo de exploraciones secuenciales, mejoras en cuanto a su rendimiento, búsquedas de texto completo, la llamada replicación síncrona, que permite el manejo de varios servidores destinados a reservas. (The PostgreSQL Global Development Group, 2017)

Para entender mejor el funcionamiento de PostgreSQL podemos ilustrarlo mediante un gráfico.

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

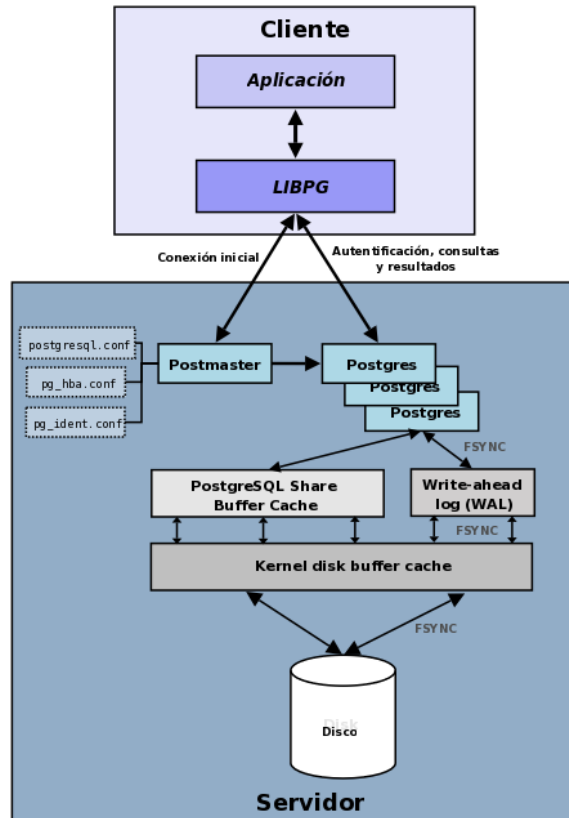


Gráfico 6: Componentes más importantes de PostgreSQL.

(The PostgreSQL Global Development Group, n.d.)

En la imagen se puede observar los componentes más importantes de PostgreSQL, entre los cuales encontramos la aplicación destinada al cliente que utiliza PostgreSQL como administrador de bases de datos, el proceso principal llamado demonio postmaster es el encargado de escuchar en un puerto/socket, también de crear los procesos hijos, que autentican las gestiones de consultas, y de enviar los resultados a las aplicaciones clientes. También podemos observar ficheros de configuración, procesos hijos con sus funciones antes mencionadas, el share buffer cache destinado para los datos en cache, WAL (Write-Ahead Logel) que se encarga de asegurar la integridad de los datos, el kernel disk buffer cache para el cache en el sistema operativo y por último en la parte final el disco donde se almacena la información para que todo funcione correctamente. (The PostgreSQL Global Development Group, n.d.)

### 2.1.2 MySQL

El sistema de gestión de bases de datos relacional MySQL, es uno de los gestores de código abierto más populares en el mercado, en especial para el empleo en aplicaciones web. Es un gestor multiusuario y multiproceso que se encuentra disponible para diferentes sistemas operativos como Linux, Windows o Mac, entre muchas otras. A un bajo costo ofrece un alto rendimiento, escalabilidad, facilidad de uso, acceso SQL y no SQL e interoperabilidad en big data. Además, cuenta con el entorno MySQL Workbench, que permite diseñar, desarrollar y gestionar bases de datos de forma productiva y segura. MySQL ofrece un soporte técnico de tiempo completo con especialistas. Además, tiene una variedad de versiones entre ellas la versión MySQL Enterprise Edition, la cual incluye herramientas de gestión y soporte técnico, ayuda a reducir el riesgo en aplicaciones críticas, cuenta con servicio en la nube y cifrado automático de datos, en tiempo real. (Jan Bodnar, 2017; Oracle, 2017). “Muchas de las organizaciones más grandes del mundo y de más rápido crecimiento usan MySQL para ahorrar tiempo y dinero, alimentando sus sitios web de gran volumen, sistemas críticos para la empresa y software empaquetado”. (“MySQL ENTERPRISE EDITION,” n.d., p. 5)

Este sistema gestor está diseñado principalmente para que se pueda cambiar según la necesidad de gestor de almacenamiento, debido a que separa a este componente del resto en su arquitectura. El gestor de almacenamiento maneja la entrada y salida de datos en la memoria secundaria. A continuación, se puede visualizar como está organizada su arquitectura.

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

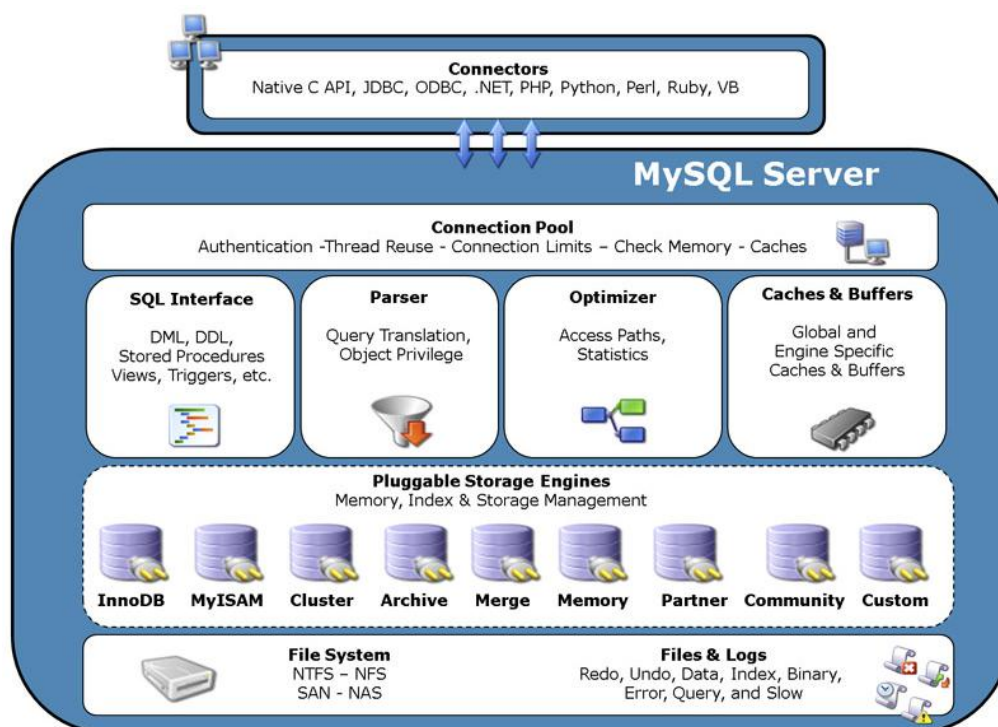


Gráfico 7: Arquitectura MySQL Server.

(“MySQL ENTERPRISE EDITION,” n.d.)

El motor de almacenamiento en MySQL tiene un conjunto de motores que se pueden enchufar o intercambiar o asignar diferentes motores a distintas bases de datos en el gestor de bases de datos o incluso un para cada tabla en una base de datos dependiendo de los requerimientos. Además, brinda la capacidad de editar el código de estos motores o desarrollar uno nuevo para tareas o aplicaciones concretas siendo independiente y compatible con el resto de servicios.

Entre otros componentes de su arquitectura están: los conectores, que son bibliotecas que permiten la conexión desde una aplicación a un servidor MySQL y realizar consultas de los datos, y el gestor de conexiones que permite mantener múltiples conexiones de los clientes, gestionando los recursos de máquina y administrar las conexiones, para evitar mantener conexiones que no se estén ocupando o saturar al servidor.

Para realizar las consultas MySQL las analiza sintácticamente y crea una representación de la cual decide qué camino tomar para mejorar la eficiencia de la consulta misma. Así mismo, antes de realizar el análisis y representación

comprueba si la consulta se encuentra en la memoria caché de consultas, donde también se encuentran almacenados sus resultados y evitar realizar de nuevo todo el proceso de ejecución. Para mantener la concurrencia de datos al realizarse consultas el motor maneja una serie de bloqueos los cuales se otorgan y liberan constantemente según las diversas políticas a tablas, usuarios, filas. (“MySQL :: MySQL 5.7 Reference Manual :: 15.11 Overview of MySQL Storage Engine Architecture,” 2017; Zaurere, 2006)

## **2.2 Sistemas de bases de datos en memoria en la actualidad.**

### **2.2.1 SAP Hana**

SAP HANA es un sistema que aprovecha la gestión de datos en memoria principal, esto le permite aumentar el rendimiento de los procesos de negocio, además las transacciones que es capaz de ejecutar se multiplican como mínimo entre 1000 y 2000 veces más veloces que de la manera convencional, ya que al acceder al disco sus tiempos son en milisegundos y cuando se accede a la memoria principal los tiempos que se obtienen son de nanosegundos, esto nos indica que los accesos son miles de veces más rápidos y se dice que su tiempo de respuesta es cero. Además, esta velocidad hace posible que aplicaciones que antes no eran posibles se vuelvan posibles, por ejemplo, brindar información al instante de procesos que realizados con la gestión de memoria en disco se tomarían mucho tiempo de proceso. (Inforges, 2013)

Otra característica por notar de HANA es que utiliza un modelo de columnas y no el tradicional modelo de filas, permitiendo que la obtención de totales sea mucho más veloz, esto nos indica que los sistemas son más simples, de manera que se puede integrar sistemas, por ejemplo, transacciones y business intelligence. (Inforges, 2013)

Para entender de una mejor forma, en la siguiente imagen se puede observar la arquitectura de SAP HANA.

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

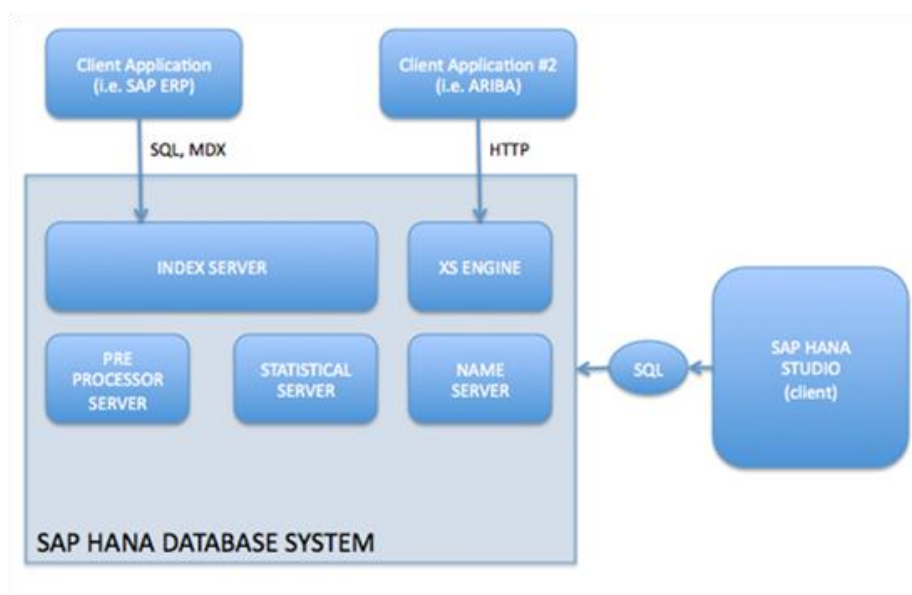


Gráfico 8: Arquitectura SAP HANA.

(Kay Somers, 2012)

- Client Applications. – Este componente de SAP HANA se puede adaptar a múltiples clientes a través de varias instancias y por medio de aplicaciones relacionadas brindar un mejor y mayor acceso a los datos.
- Index Server. – Aquí se alojan los datos reales y los motores para la gestión de los datos.
- Pre-Processor Server. - Para entender el funcionamiento del Pre-Processor Server debemos tener en cuenta que el index server utiliza al Pre-Processor Server para el análisis de los datos y extraer esos datos mediante una función de búsqueda.
- Statistical Server. - Es el encargado de analizar el estado actual, las métricas y el rendimiento de todos los componentes de SAP HANA.
- Name Server. - Aquí se registra la topología de la base de datos para disminuir los tiempos asociados a la indexación.

También podemos mencionar a la versión más reciente SAP HANA 2, que fue lanzada oficialmente el 30 de noviembre de 2016, pero esto no indica que la primera versión quede obsoleta, de hecho, la primera versión recibirá un mantenimiento el segundo semestre de 2017. Entre las novedades que trae esta última versión tenemos, mejoras en alta disponibilidad y seguridad, una

aplicación web basada en SAP HANA que da la posibilidad a los clientes de apreciar el impacto de nuevas tecnologías en las organizaciones, una mejor integración de información adicional, entre otras, esto significa un adelanto para la toma de decisiones y los resultados del negocio. También es importante mencionar que SAP HANA y SAP HANA 2 son completamente compatibles en todas sus funciones y no exige cambios en la arquitectura. (Andreas Schmitz, 2016)

### **2.2.2 Oracle TimesTen In-Memory Database**

La base de datos en memoria Oracle TimesTen es una de datos relacional que busca brindar respuesta instantánea y un alto rendimiento a aplicaciones en tiempo real mediante el aprovechamiento de la memoria física, se puede emplear para aplicaciones donde el tiempo de respuesta es crítico ya sean OLTP o aplicaciones analíticas de apoyo a toma de decisiones. (“Oracle Data sheet | TimesTen In-Memory Database,” 2014)

TimesTen permite mantener persistencia y recuperación de datos, también aislamiento multiusuario y replicación en servidores para aumentar la disponibilidad y además el acceso a los datos se puede realizar mediante interfaces SQL estándar.

Utiliza puntos de control, que son registros de transacciones realizadas en la base de datos, escritos en una imagen de disco para conseguir durabilidad. Dichos registros, se pueden escribir según una configuración de tiempo predeterminado periódicamente en el disco.

Por otro lado, utiliza conexiones convencionales cliente / servidor para conectarse con la base de datos, sin embargo, permite mejorar dicha conexión incluyendo a TimesTen en las aplicaciones, evitando así, la comunicación de procesos y sobrecargas en la red.

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

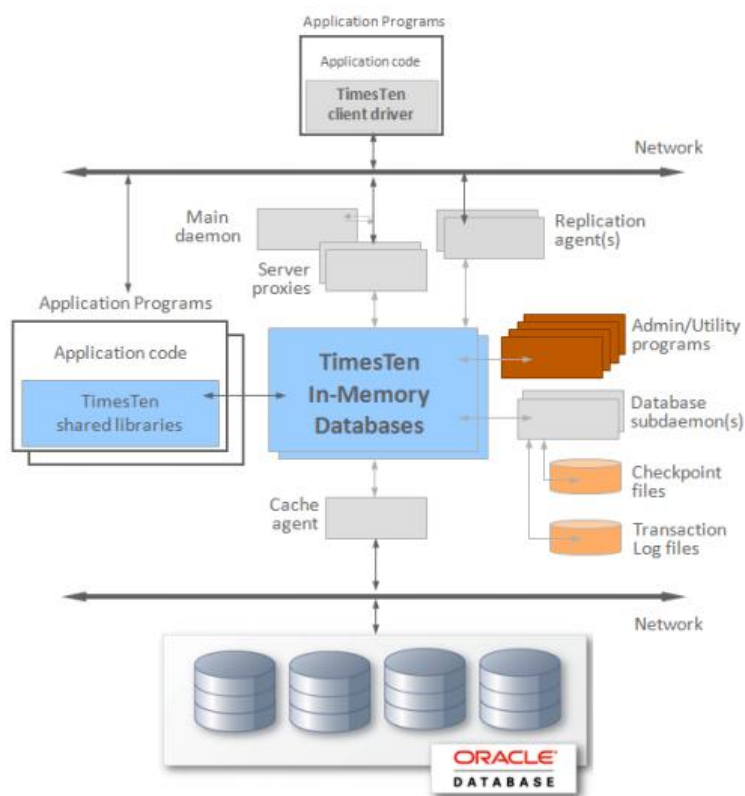


Gráfico 9: Componentes de Oracle TimesTen In-Memory Database.

(“Oracle White Paper | Extreme Performance Using Oracle TimesTen In-Memory Database,” 2014)

En su arquitectura, TimesTen incluye un conjunto de bibliotecas compartidas que permiten a los desarrolladores incluir a TimesTen como parte de los procesos en sus aplicaciones, así se puede obtener un mejor rendimiento al tener una aplicación directamente vinculada en comparación con conexión cliente / servidor.

TimesTen emplea un algoritmo llamado MicroLogging en sus bibliotecas para disminuir la vulnerabilidad y protegerse de fallos, permitiendo que otras aplicaciones se ejecuten sin interrupción.

Mantiene las bases de datos en memoria con todos los datos de usuarios, índices, catálogos, etc., para que muchas aplicaciones puedan acceder a estos o a diferentes bases de datos.

Posee procesos en segundo plano, denominados daemon, en el sistema para ofrecer servicios que detecten el encendido, el apagado y fallos de aplicación, y también, que proporcionen puntos de verificación, mantener sub-instancias

separadas para cada base de datos y un daemon principal para instanciar a TimesTen.

Cuenta con programas administrativos para brindar diferentes utilidades a los usuarios como copias de seguridad, restauración, migración de bases de datos, entre otros. (“Oracle White Paper | Extreme Performance Using Oracle TimesTen In-Memory Database,” 2014)

## **2.3 Sistemas de bases de datos mixta en la actualidad.**

### **2.3.1 Oracle Database 12c con Oracle Database In-Memory**

La opción Oracle Database In-Memory incluida en Oracle Database 12c, ofrece la capacidad de soportar cargas de trabajo con un alto rendimiento para transacciones y análisis debido a que presenta una arquitectura doble, de esta manera las tablas se pueden representar simultáneamente en disco y memoria principal. Por un lado, cuenta con un formato de filas para el procesamiento de transacciones en línea, y un formato columnar en memoria enfocado en el procesamiento analítico. También, cuenta con un optimizador de consultas que ayuda a dirigir las consultas a dichos enfoques, columnar o en formato de filas. La información que exige un mayor rendimiento es aquella que se debe seleccionar para ubicarse en memoria y generar un formato columnar, claro que si la base de datos es pequeña o se cuenta con la memoria suficiente se puede mantener todos los objetos de dicha base de datos, en memoria. Así como también, las bases de datos existentes pueden actualizarse a la opción de en memoria sin tener que alterar el hardware o software y beneficiarse de su rendimiento con algunos objetos en formato columnar y mantener la base de datos en disco. (Kyte, 2015; Rivenes, Colgan, & Marwah, 2017)

## Rendimiento optimizado de Transacciones y Querys

### Databases en formato filas versus formato columnar



Gráfico 10: Rendimiento optimizado de Transacciones y Querys.

(Espin, 2014).

### 2.3.2 Microsoft SQL Server 2014 y 2016

Cuando hablamos de los gestores de bases de datos mixtos en la actualidad, podemos mencionar como ejemplo SQL Server de Microsoft, este gestor es uno de los motores más usados últimamente, que por medio de OLTP en memoria optimiza y mejora considerablemente el rendimiento de escenarios de datos transitorios, carga de datos, ingesta de datos y procesamiento de transacciones, esto es posible por su nuevo formato de almacenamiento. El manejo de las tablas o los procesos almacenados que son llevados a OLTP en memoria se vuelve posible a partir de la versión 2014 de SQL Server, pero SQL Server 2016 lo mejora significativamente.

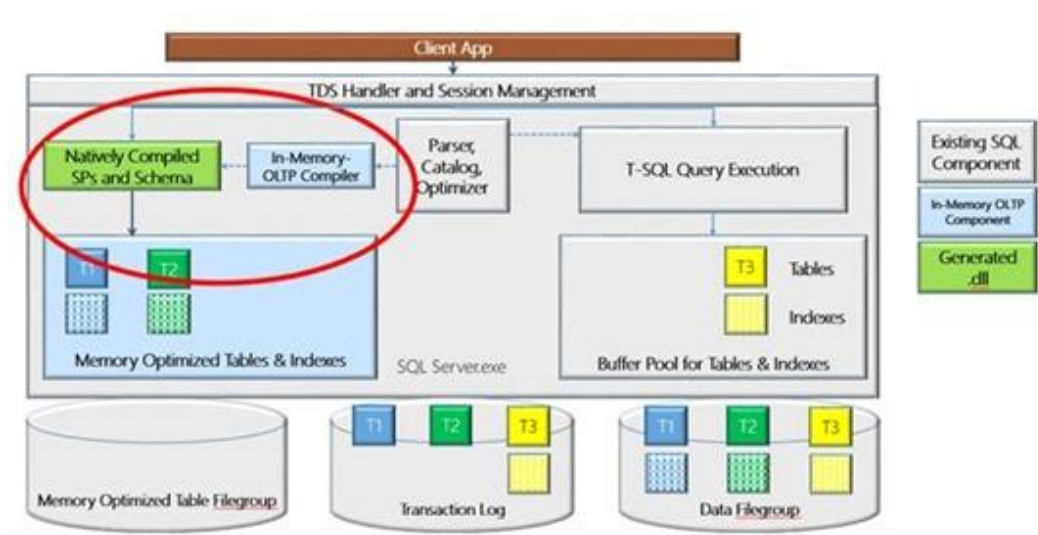


Gráfico 11: Estructura de funcionamiento SQL Server 2014.

(MSFT, 2014).

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

Centrándonos en la última versión del año 2016 es importante mencionar como se determina si una tabla o un procedimiento almacenado se debe pasar a OLTP en memoria y esto es mediante el asesor de optimización de memoria, ya que este permite identificar las características usadas en una tabla basada en disco que no se admiten en tablas con optimización para memoria y, además permite migrar una tabla y los datos a optimización para memoria (Microsoft, n.d.).

Otra de las novedades que trae esta versión es el funcionamiento en arquitectura de 64 bits, pero es importante mencionar que no todas las funcionalidades se ejecutan en 64 bits. En cuanto al esquema de funcionamiento de SQL server 2016 podemos observar a Hekaton mejorado en la versión 2016.

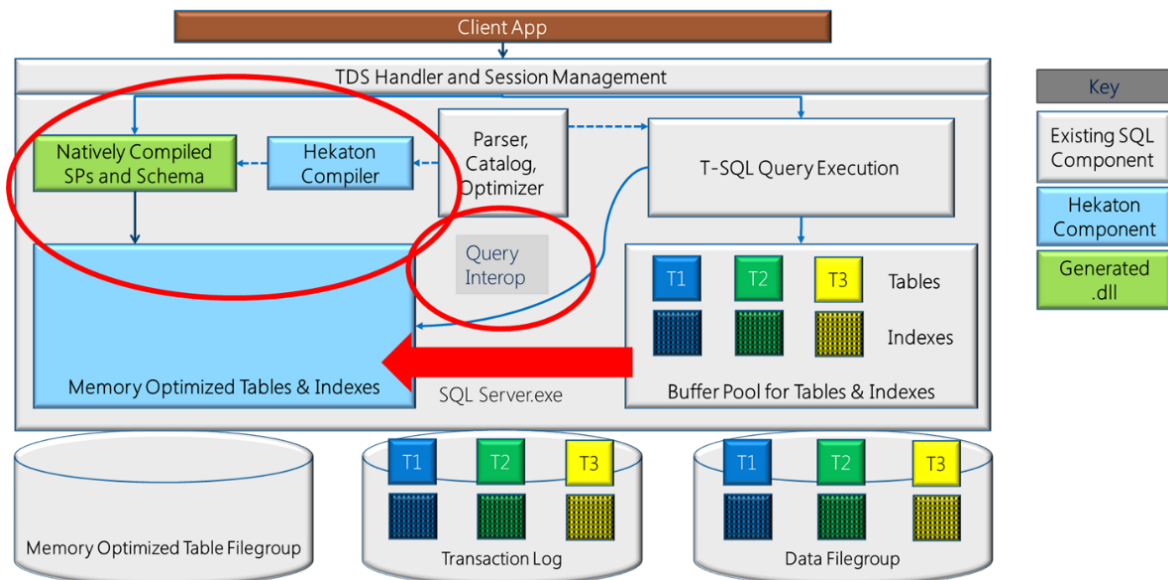


Gráfico 12: Estructura de funcionamiento SQL Server 2016.

(Prakash Karia, 2013).

Para detallar que es Hekaton podemos decir que es un motor de base de datos que fue perfeccionado para datos localizados en memoria y cargas de trabajo OLTP. Hekaton es una funcionalidad integrada en el gestor y no aparte, las tablas de Hekaton son transaccionales y duraderas, se accede utilizando T-SQL de igual forma que las tablas regulares de SQL Server. Es importante mencionar que las consultas que se realicen pueden hacer referencia a las tablas Hekaton y también a las tablas regulares, así también la actualización de

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

las mismas se puede realizar normalmente a través de una transacción (Diaconu et al., n.d.).

## 2.4 Cuadro Resumen Gestores de Bases de datos.

PostgreSQL	MySQL	SAP HANA	Oracle TimesTen	Oracle Database	Microsoft SQL Server
<p>Muy popular, avanzado y potente. Gestor de código Libre. Licencia BSD. Maneja modelo cliente/servidor. Usa multiprocesos. Última versión 9.6. Gestor Relacional.</p>	<p>Gran popularidad. Gestor de código abierto. Multiusuario y Multiproceso. Varios Sistemas Operativos. Bajo Coste. Alto rendimiento y escalabilidad. Cuenta con entorno MySQL Workbench. Variedad en versiones. Motor de almacenamiento separado.</p>	<p>Gestión en memoria principal. Gran velocidad de ejecución de consultas. Modelo columnar. Simplicidad de sistemas. Permite integración de sistemas. Versión SAP HANA 2. Maneja modelo cliente/servidor.</p>	<p>Base de datos relacional. Inclusión y alto rendimiento en aplicaciones en tiempo real. Gestión en memoria. Persistencia y Recuperación de datos. Multiusuario y replicación. Puntos de control. Brinda programas administrativos.</p>	<p>Opción Oracle Database In-Memory. Formato en filas - procesamiento de transacciones. Formato columnar - procesamiento analítico. Optimizador de consultas. Migración a memoria Última versión 12c.</p>	<p>Entre los más usados. Gestión en memoria principal. Asesor de optimización de memoria. Migración a memoria. Última versión 2016.</p>

Tabla 3: Cuadro Resumen Gestores de Bases de datos.

(Elaboración Propia).

### 3 Capítulo: Selección de indicadores y ambiente de evaluación

#### 3.1 Características principales.

Para realizar las pruebas de evaluación en los gestores de bases de datos, para su posterior comparación, utilizaremos un ordenador de escritorio con las siguientes especificaciones técnicas.

- Procesador: Intel(R) Core™ i7-3770 CPU @ 3.40GHz, procesador x64.
- Memoria RAM DIMM 1: 4 GB DDR3-1333 DDR3 SDRAM.
- Memoria RAM DIMM 2: 4 GB DDR3-1333 DDR3 SDRAM.
- Disco de almacenamiento: 1TB 7200rpm Sata.
- Tarjeta Gráfica (Integrada): Placa de video Intel(R) HD Graphics 4000 (2112 MB).
- Sistema Operativo: Windows 10 Pro de 64 bits. 2017-06 Actualización acumulativa para Windows 10 Versión 1703 para sistemas basados en x64 (KB4022725)

Dentro del mismo se empleará una herramienta para la creación de 3 máquinas virtuales, que nos permitirán instalar y realizar las pruebas bajo un ambiente similar, sin que cada uno de los procesos de los gestores de bases de datos interrumpen a otros procesos, con fin de eliminar las posibles alteraciones a los resultados.

Utilizaremos máquinas virtuales con las siguientes especificaciones.

- Memoria RAM asignada: 6 GB.
- Disco de almacenamiento asignado: 80 GB.

En cuanto a la versión de sistema operativo que se maneja en las 3 máquinas virtuales tenemos:

- Sistema Operativo: Windows 10 Pro de 64 bits, versión 1703.

Esta versión fue la más actualizada en el momento de la creación de la máquina virtual.

Opción de mantenimiento	Version	OS build	Fecha de disponibilidad	Fecha de última revisión
Rama actual (CB)	1703	15063.447	4/11/2017	6/27/2017

Tabla 4: Versiones de Windows 10 por opción de mantenimiento.

(© Microsoft, 2017)

OS build	Fecha de disponibilidad	Opción de mantenimiento	Artículo de KB
15063.413	6/13/2017	CB	<a href="#">KB 4022725</a>

Tabla 5: Versión 1703 (OS build 15063).

(© Microsoft, 2017)

### 3.2 Selección de sistemas de base de datos.

En este estudio, se manejará a los gestores de bases de datos mixtas también como administradores de bases de datos en disco. Inicialmente, utilizando su forma tradicional de gestión de bases de datos, posteriormente asignando un tamaño mediano de memoria para los objetos más críticos de la base de datos y finalmente, asignando todo el espacio de memoria disponible a toda la base de datos. Esta decisión se ha tomado, con el objetivo de tener un mejor ambiente de comparación. De esta forma, obtendremos resultados que nos permitirán comparar al mismo gestor en sus diferentes ambientes y poder identificar sus beneficios y desempeño en cada uno de ellos.

Por otro lado, se pondrá a prueba un gestor propio de la base de datos para trabajar con la base de datos completamente en la memoria física, permitiéndonos contrastar con los otros gestores cuando se encuentren configurados para operar solo en la memoria principal.

#### 3.2.1 Selección de Microsoft SQL Server 2016.

La elección de Microsoft SQL Server se debe a que como se explicó anteriormente, este sistema gestor permite la gestión de memoria de forma tradicional, que usan los manejadores de bases de datos en disco. Además, permite trabajar con la opción en memoria, la cual ofrece la posibilidad de asignar a distintos objetos de la base de datos a la memoria RAM, según se requiera o se disponga de dicha memoria RAM.

En cuanto a las principales razones para la elección de Microsoft SQL Server tenemos:

- Microsoft SQL Server cuenta con el manejo de la información entre memorias de una forma mixta, se puede asignar el espacio a utilizar en la memoria principal.

- Sus bases de datos permiten manejar modelos relacionales, esto es importante ya que el modelo de la base de datos para la comparativa será un modelo relacional.
- SQL Server 2016 brinda una interfaz de administración de base de datos que facilita las operaciones sobre las bases de datos por medio de SQL Server Management Studio (SSMS).
- Su lenguaje para las sentencias es el conocido como Transact-SQL (TSQL), otro punto importante ya que las sentencias a utilizar pueden ser manejadas sin problemas con este lenguaje.
- Es completamente compatible y funcional en ambiente Windows, esto porque se usará este sistema operativo para la comparativa.
- El ordenador que se empleará para las pruebas cumple con los requisitos de funcionamiento de Microsoft SQL server.
- Por último, Microsoft SQL server se encuentra disponible en una versión de evaluación de prueba gratuita por 180 días, para cualquier persona en su página oficial.

### **3.2.2 Selección de Oracle Database 12c (Release 2).**

La elección de Oracle Database 12c se ha dado debido a que igualmente como en el caso de SQL Server 2016, este sistema gestor permite la gestión de memoria de forma tradicional y además con la opción en memoria.

En cuanto a las principales razones para la elección de Oracle Database 12c, tenemos:

- Oracle Database 12c permite el manejo de la información entre memorias de una forma mixta.
- Sus bases de datos permiten manejar modelos relacionales.
- Su lenguaje para las sentencias es el conocido como PL/SQL, lenguaje incrustado de Oracle que nos permite trabajar sin problemas con las consultas a realizar para las pruebas.
- Oracle Database 12c brinda una interfaz de administración de base de datos que facilita las operaciones sobre las bases de datos por medio de Oracle SQL Developer.
- Es completamente compatible y funcional en ambiente Windows.

- El ordenador que se empleará para las pruebas cumple con los requisitos de funcionamiento de Oracle Database 12c.
- Por último, Oracle Database 12c se encuentra disponible en una versión de descarga gratuita en su página oficial bajo el acuerdo de licencia OTN (Oracle Technology Network).

### **3.2.3 Selección de Oracle TimesTen.**

En cuanto al software elegido como gestor de bases de datos en memoria tenemos a TimesTen de Oracle, las principales razones para su elección son:

- Oracle TimesTen fue desarrollada desde cero como una base de datos en memoria.
- Permite el almacenamiento de bases de datos relacionales basado en filas, factor de importancia ya que el modelo de la base de datos para la comparativa será un modelo relacional.
- Maneja el lenguaje PL/SQL y utiliza interfaces SQL estándar para el acceso a datos y ejecución de sentencias.
- Es completamente compatible y funcional en ambiente Windows, sistema operativo que se empleará en la comparativa.
- Oracle TimesTen se puede conectar con la herramienta Oracle SQL Developer que ayuda a la gestión de base de datos de la misma forma que la versión de Oracle Database 12c.
- El ordenador que se empleará cuenta con una buena cantidad de memoria RAM para las pruebas. La memoria RAM necesaria depende del tamaño de la base de datos con la que se vaya a trabajar, ya que todos los datos deben residir en la memoria principal.
- En la página de Oracle se encuentra suficiente documentación como tutoriales y demos para un buen manejo de la herramienta.
- El ordenador que se empleará para las pruebas cumple con los requisitos de funcionamiento de Oracle TimesTen.
- Por último, Oracle TimesTen se encuentra disponible en una versión de descarga gratuita bajo el acuerdo de licencia OTN (Oracle Technology Network) en su página oficial.

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

Para finalizar los criterios de selección, se tuvo en cuenta los rankings de las bases de datos, que arrojaron estudios realizados por BD-Engines y Forrester Wave, que catalogan a Oracle y Microsoft dentro del grupo de líderes de gestores de bases de datos.






















Rank			DBMS	Database Model	Score		
Jul 2017	Jun 2017	Jul 2016			Jul 2017	Jun 2017	Jul 2016
1.	1.	1.	Oracle  	Relational DBMS	1374.88	+23.11	-66.65
2.	2.	2.	MySQL  	Relational DBMS	1349.11	+3.80	-14.18
3.	3.	3.	Microsoft SQL Server  	Relational DBMS	1226.00	+27.03	+33.11
4.	4.	 5.	PostgreSQL  	Relational DBMS	369.44	+0.89	+58.28
5.	5.	 4.	MongoDB  	Document store	332.77	-2.23	+17.77
6.	6.	6.	DB2 	Relational DBMS	191.25	+3.74	+6.17
7.	7.	 8.	Microsoft Access	Relational DBMS	126.13	-0.42	+1.23
8.	8.	 7.	Cassandra 	Wide column store	124.12	-0.00	-6.58
9.	9.	 10.	Redis 	Key-value store	121.51	+2.63	+13.48
10.	 11.	 11.	Elasticsearch 	Search engine	115.98	+4.42	+27.36

Tabla 6: DB-Engines Ranking.

(DB-Engines, 2017)

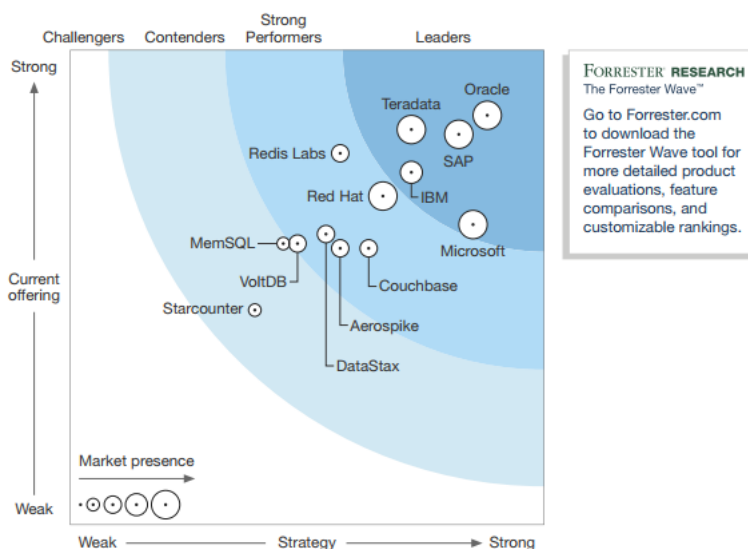


Gráfico 13: Forrester Wave: In-Memory Databases, 2017.

(Forrester Wave, 2017)

### 3.3 Materiales y condiciones para la realización de pruebas.

#### 3.3.1 Diseño de modelo de base de datos.

Para la realización de las pruebas se diseñó una base de datos simple con millones de registros que permita realizar las pruebas de rendimiento adecuadas al análisis que se realizará.

### 3.3.1.1 Modelo Relacional

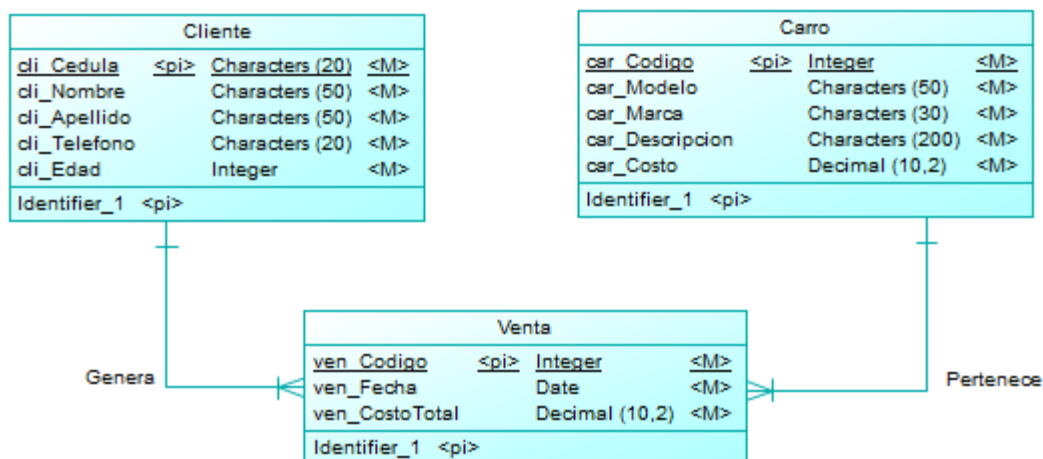


Gráfico 14: Modelo Relacional de la Base de datos de Prueba.

(Elaboración Propia).

### 3.3.1.2 Modelo Físico

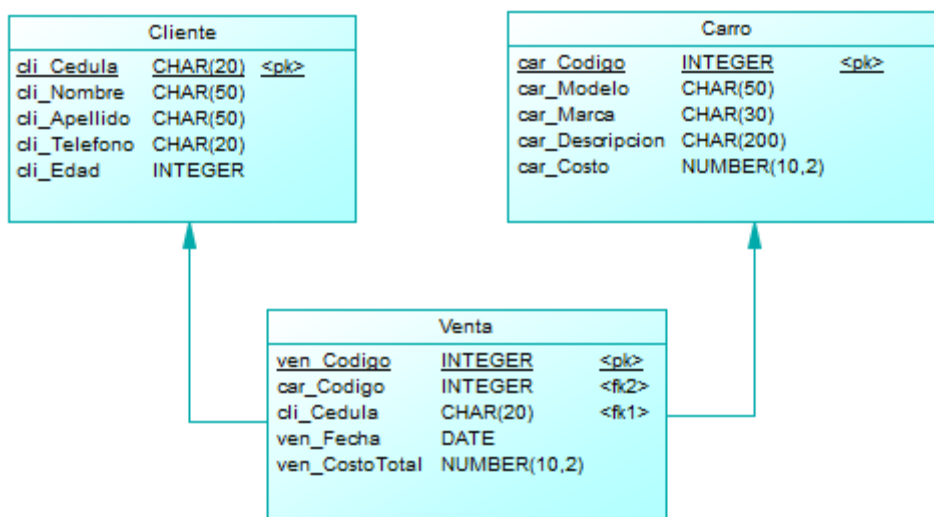


Gráfico 15: Modelo Físico de la Base de datos de Prueba.

(Elaboración Propia).

## 3.3.2 Software de ayuda.

### 3.3.2.1 RedGate (SQL Data Generator)

Para poblar la base de datos con millones de registros validos se empleó la herramienta SQL Data Generator. RedGate, ofrece muchas herramientas adicionales que trabajan de la mano con SQL Server, como es el caso del SQL Data Generator, herramienta que nos permite crear grandes volúmenes de

datos reales en SQL Server Management Studio. (Red Gate Software Ltd, 2017)

### **3.4 Selección de indicadores para la realización del análisis benchmark.**

En este estudio se analizaron indicadores de trabajos que fueron realizados como comparativas entre bases de datos, para encontrar una selección adecuada de los mismos, que proporcione operaciones estándar dentro de las bases de datos, obteniendo como resultado los siguientes indicadores.

#### **3.4.1 Coincidencias exactas (Q1) (Q2)**

Este indicador permite evaluar la búsqueda de coincidencias de caracteres dado un patrón específico. En SQL el operador LIKE permite realizar dichas coincidencias.

#### **3.4.2 Acceso ordenado (Q3) (Q4)**

Este indicador permite evaluar el desempeño de una enumeración ordenada de resultados, ya sean de forma ascendente o descendente dependiente de un atributo especificado. En SQL el operador ORDER BY permite realizar dicha acción.

#### **3.4.3 Casting (Q5)**

Este indicador permite evaluar el desempeño de la conversión de un tipo de dato en otro tipo de dato. En SQL el operador CAST permite realizar dicha acción.

#### **3.4.4 Expresiones regulares (Q6) (Q7)**

Este indicador permite evaluar el desempeño cuando se maneja búsquedas usando cadenas de caracteres que definen un patrón para la búsqueda, que se compone de literales y meta caracteres. En SQL el operador LIKE permite realizar búsquedas con expresiones regulares.

#### **3.4.5 Encontrar referencias (Q8) (Q9)**

Este indicador permite evaluar el desempeño de una consulta en la cual intervengan campos referenciados o relacionados con campos de otras tablas. En SQL se puede realizar esta acción por medio del cruce de tablas.

#### **3.4.6 Select Anidado (Q10)**

Este indicador permite evaluar el comportamiento de una consulta en la cual intervengan consultas secundarias que proporcionen información a la consulta principal. En SQL se puede realizar esta acción mediante el uso de SELECT de forma anidada.

#### **3.4.7 Joins (Q11) (Q12) (Q13)**

Este indicador permite evaluar el comportamiento de una consulta en la cual se emplea la cláusula JOIN que es utilizada para combinar filas de dos o más tablas, en función de una columna relacionada entre ellas. En SQL se puede emplear RIGHT, LEFT, FULL, SELF, INNER JOIN, según se requiera.

#### **3.4.8 Búsqueda por índice (Q14)**

Este indicador permite evaluar el comportamiento de la creación de un índice sobre un campo para optimizar la consulta hacia dicho campo. En SQL se puede crear índices con la sentencia CREATE INDEX para su posterior uso.

#### **3.4.9 Agregación (Q15) (Q16)**

Este indicador permite evaluar el desempeño de una consulta que emplee funciones de agregación, las cuales realizan operaciones sobre un conjunto de datos. En SQL se pueden emplear distintas funciones de agregación entre las más comunes están: COUNT, MIN, MAX, SUM y AVG.

#### **3.4.10 Inserción (Q17)**

Este indicador permite evaluar el comportamiento de la inserción de un nuevo registro a una tabla. En SQL se puede realizar inserciones de registros con la sentencia INSERT INTO.

#### **3.4.11 Actualización (Q18) (Q19)**

Este indicador permite evaluar el comportamiento de la actualización de uno o varios registros en uno o varios campos, establecidos por una condición. En SQL se puede realizar actualizaciones de registros con la sentencia UPDATE.

#### **3.4.12 Eliminación (Q20) (Q21)**

Este indicador permite evaluar el comportamiento de la eliminación de uno o varios registros establecidos por una condición. En SQL se puede realizar eliminaciones de registros con la sentencia DELETE.

### 3.5 Elaboración de Matrices para el análisis benchmark.

Esta matriz fue diseñada para la presentación de la forma (En Disco) del gestor de bases de datos, permitiéndonos la presentación de los 5 tiempos recolectados por cada consulta, liberando memoria en cada uno de ellos y con la base de datos en disco. Además, se muestra el número de registros que arrojan como resultado en cada consulta y el correspondiente promedio de los 5 tiempos.

		Nombre del Gestor						
Indicadores	N° Query	En Disco					P	N° Registros
		1	2	3	4	5		
Coincidencias exactas	(Q1)							
	(Q2)							
Acceso ordenado	(Q3)							
	(Q4)							
Casting	(Q5)							
Expresiones regulares	(Q6)							
	(Q7)							
Encontrar referencias	(Q8)							
	(Q9)							
Select anidado	(Q10)							
Joins	(Q11)							
	(Q12)							
	(Q13)							
Búsqueda por índice	(Q14)							
Agregación	(Q15)							
	(Q16)							
Inserción	(Q17)							
Actualización	(Q18)							
	(Q19)							
Eliminación	(Q20)							
	(Q21)							

Tabla 7: Modelo Matriz para la comparativa de datos en Disco.

(Elaboración Propia).

Esta matriz fue diseñada para la presentación de la forma (Mixta) del gestor de bases de datos, permitiéndonos la presentación de los 5 tiempos recolectados en las consultas que son factibles su ejecución de forma mixta, como en el caso anterior los tiempos se toman liberando memoria en cada uno de ellos. Además, se muestra el número de registros que arrojan como resultado cada consulta y el correspondiente promedio de los 5 tiempos.

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

		Nombre del Gestor						N° Registros
Indicadores	N° Query	Mixtas					P	
		1	2	3	4	5		
Coincidencias exactas	(Q1)	-						
	(Q2)							
Acceso ordenado	(Q3)	-						
	(Q4)							
Casting	(Q5)	-						
Expresiones regulares	(Q6)	-						
	(Q7)							
Encontrar referencias	(Q8)							
	(Q9)							
Select anidado	(Q10)							
Joins	(Q11)							
	(Q12)							
	(Q13)							
Búsqueda por índice	(Q14)	-						
Agregación	(Q15)	-						
	(Q16)	-						
Inserción	(Q17)	-						
Actualización	(Q18)	-						
	(Q19)	-						
Eliminación	(Q20)	-						
	(Q21)	-						

Tabla 8: Modelo Matriz para la comparativa de datos Mixta.

(Elaboración Propia).

Esta matriz fue diseñada para la presentación de la forma (En Memoria) del gestor de bases de datos, permitiéndonos la presentación de los 5 tiempos recolectados por cada consulta, liberando memoria en cada uno de ellos y con la base de datos en memoria, además se muestra el número de registros que arrojan como resultado cada consulta y el correspondiente promedio de los 5 tiempos.

		Nombre del Gestor						N° Registros
Indicadores	N° Query	En Memoria					P	
		1	2	3	4	5		
Coincidencias exactas	(Q1)							
	(Q2)							
Acceso ordenado	(Q3)							
	(Q4)							
Casting	(Q5)							
Expresiones regulares	(Q6)							
	(Q7)							
Encontrar referencias	(Q8)							
	(Q9)							
Select anidado	(Q10)							
Joins	(Q11)							
	(Q12)							
	(Q13)							
Búsqueda por índice	(Q14)							
Agregación	(Q15)							
	(Q16)							
Inserción	(Q17)							
Actualización	(Q18)							
	(Q19)							
Eliminación	(Q20)							
	(Q21)							

Tabla 9: Modelo Matriz para la comparativa de datos en Memoria.

(Elaboración Propia).

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

Esta matriz fue diseñada para la presentación de cada forma del gestor correspondiente (Disco, Mixta y Memoria) dándonos la observación global de las calificaciones obtenidas por las consultas.

Nombre del Gestor				
Indicadores	N° Query	En Disco	Mixtas	En Memoria
Coincidencias exactas	(Q1)			
	(Q2)			
Acceso ordenado	(Q3)			
	(Q4)			
Casting	(Q5)			
Expresiones regulares	(Q6)			
	(Q7)			
Encontrar referencias	(Q8)			
	(Q9)			
Select anidado	(Q10)			
Joins	(Q11)			
	(Q12)			
	(Q13)			
Búsqueda por índice	(Q14)			
Agregación	(Q15)			
	(Q16)			
Inserción	(Q17)			
Actualización	(Q18)			
	(Q19)			
Eliminación	(Q20)			
	(Q21)			
Puntaje Final				

Tabla 10: Modelo Matriz de calificación para la comparativa en cada Base de Datos.

(Elaboración Propia).

Esta matriz fue diseñada para la presentación de cada forma del gestor correspondiente (Disco, Mixta y Memoria) dándonos la observación global de los promedios de todas las consultas, para su posterior análisis.

Nombre del Gestor				
Indicadores	N° Query	En Disco	Mixtas	En Memoria
Coincidencias exactas	(Q1)		-	
	(Q2)			
Acceso ordenado	(Q3)		-	
	(Q4)		-	
Casting	(Q5)		-	
Expresiones regulares	(Q6)		-	
	(Q7)			
Encontrar referencias	(Q8)			
	(Q9)			
Select anidado	(Q10)			
Joins	(Q11)			
	(Q12)			
	(Q13)			
Búsqueda por índice	(Q14)		-	
Agregación	(Q15)		-	
	(Q16)		-	
Inserción	(Q17)		-	
Actualización	(Q18)		-	
	(Q19)		-	
Eliminación	(Q20)		-	
	(Q21)		-	

Tabla 11: Modelo Matriz de promedios para la comparativa en cada Base de Datos.

(Elaboración Propia).

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

Esta matriz fue diseñada para la presentación de cada forma del gestor correspondiente (Disco, Mixta y Memoria) dándonos la observación global de los tiempos obtenidos en la segunda ejecución de las consultas sin liberar memoria.

Nombre del Gestor					
Indicadores	N° Query	Sin Liberar Memoria			N° Registros
		En Disco	Mixtas	En Memoria	
Coincidencias exactas	(Q1)		-		
	(Q2)				
Acceso ordenado	(Q3)		-		
	(Q4)		-		
Casting	(Q5)		-		
Expresiones regulares	(Q6)		-		
	(Q7)				
Encontrar referencias	(Q8)				
	(Q9)				
Select anidado	(Q10)				
Joins	(Q11)				
	(Q12)				
	(Q13)				
Búsqueda por índice	(Q14)		-		
Agregación	(Q15)		-		
	(Q16)		-		
Inserción	(Q17)		-		
Actualización	(Q18)		-		
	(Q19)		-		
Eliminación	(Q20)	-	-	-	
	(Q21)	-	-	-	

Tabla 12: Modelo Matriz para la comparativa sin liberar memoria en cada Base de Datos.

(Elaboración Propia).

## 4 Capítulo: Evaluación y Análisis benchmark

### 4.1 Pruebas de rendimiento SQL Server 2016.

Para la toma de datos en SQL Server 2016 se tomó en cuenta que los valores de tiempo indicados por el compilador coincidan con los valores tomados externamente. Además, se manejaron 3 decimales para mayor exactitud en los cálculos.

#### 4.1.1 SQL Server 2016 en disco – Evaluación 1.

SQL Server 2016								
Indicadores	N° Query	En Disco						N° Registros
		1	2	3	4	5	P	
Coincidencias exactas	(Q1)	00:32,610	00:29,874	00:29,256	00:29,281	00:30,868	00:30,378	16389
	(Q2)	00:33,695	00:37,480	00:39,031	00:39,937	00:42,921	00:38,613	36121
Acceso ordenado	(Q3)	02:12,766	02:19,110	02:14,547	02:14,187	02:11,172	02:14,356	2000000
	(Q4)	02:05,088	01:55,767	01:49,516	01:50,453	01:49,093	01:53,983	2000000
Casting	(Q5)	01:43,422	01:51,311	01:50,875	01:48,094	01:48,501	01:48,441	1574113
Expresiones regulares	(Q6)	00:13,238	00:13,129	00:15,085	00:15,531	00:16,314	00:14,659	981593
	(Q7)	00:58,703	00:53,328	00:51,707	00:54,625	00:52,945	00:54,262	1675642
Encontrar referencias	(Q8)	01:01,547	01:00,125	00:57,656	01:00,359	01:01,078	01:00,153	1980184
	(Q9)	01:39,859	01:36,811	01:35,844	01:36,308	01:31,643	01:36,093	15
Select anidado	(Q10)	02:17,657	02:09,953	02:08,609	02:10,048	02:05,595	02:10,372	295847
Joins	(Q11)	01:18,734	01:16,062	01:17,249	01:17,266	01:16,216	01:17,105	4000000
	(Q12)	00:58,016	00:56,440	00:57,172	00:58,297	00:55,782	00:57,141	1980184
	(Q13)	01:57,422	01:55,859	01:52,753	01:54,904	01:54,430	01:55,074	2000000
Búsqueda por índice	(Q14)	00:27,410	00:29,249	00:26,812	00:27,012	00:27,766	00:27,650	116733
Agregación	(Q15)	00:08,510	00:09,931	00:09,992	00:10,093	00:09,782	00:09,662	20089
	(Q16)	01:23,491	01:26,829	01:27,359	01:24,494	01:24,612	01:25,357	15
Inserción	(Q17)	00:00,337	00:00,313	00:00,401	00:00,290	00:00,326	00:00,333	100
Actualización	(Q18)	09:00,126	09:02,001	09:01,545	09:00,981	09:03,002	09:01,531	133408
	(Q19)	26:23,988	26:14,015	26:23,506	26:22,881	26:05,714	26:18,021	2000000
Eliminación	(Q20)	01:37,079	01:24,986	01:41,103	01:38,462	01:37,057	01:35,737	1692489
	(Q21)	01:35,422	01:43,500	01:39,560	01:42,973	01:43,430	01:40,977	2000000

Tabla 13: Matriz de toma de datos SQL Server 2016 en disco.

(Elaboración Propia).

### 4.1.2 SQL Server 2016 mixta – Evaluación 2.

SQL Server 2016								
Indicadores	N° Query	Mixtas						N° Registros
		1	2	3	4	5	P	
Coincidencias exactas	(Q1)	-						16389
	(Q2)	00:29,954	00:33,516	00:32,640	00:29,516	00:27,290	00:30,583	36121
Acceso ordenado	(Q3)	-						2000000
	(Q4)	-						2000000
Casting	(Q5)	-						1574113
Expresiones regulares	(Q6)	-						981593
	(Q7)	00:40,485	00:43,011	00:41,938	00:42,484	00:43,830	00:42,350	1675642
Encontrar referencias	(Q8)	00:53,016	00:52,345	00:48,594	00:48,392	00:52,532	00:50,976	1980184
	(Q9)	01:33,641	01:42,422	01:32,639	01:28,132	01:23,282	01:32,023	15
Select anidado	(Q10)	02:47,985	02:07,578	02:20,189	02:38,036	02:17,563	02:26,270	295847
Joins	(Q11)	00:55,391	01:01,005	01:01,719	01:07,578	01:06,688	01:02,476	4000000
	(Q12)	00:40,906	00:48,829	00:45,906	00:48,050	00:47,703	00:46,279	1980184
	(Q13)	01:48,860	01:44,141	01:53,485	01:48,469	01:53,985	01:49,788	2000000
Búsqueda por índice	(Q14)	-						116733
Agregación	(Q15)	-						20089
	(Q16)	-						15
Inserción	(Q17)	-						100
Actualización	(Q18)	-						133408
	(Q19)	-						2000000
Eliminación	(Q20)	-						1692489
	(Q21)	-						2000000

Tabla 14: Matriz de toma de datos SQL Server 2016 mixta.

(Elaboración Propia).

### 4.1.3 SQL Server 2016 en memoria – Evaluación 3.

SQL Server 2016								
Indicadores	N° Query	En Memoria						N° Registros
		1	2	3	4	5	P	
Coincidencias exactas	(Q1)	00:02,141	00:02,252	00:02,155	00:03,093	00:02,448	00:02,418	16389
	(Q2)	00:10,109	00:10,375	00:09,860	00:10,391	00:10,175	00:10,182	36121
Acceso ordenado	(Q3)	00:26,640	00:23,297	00:22,453	00:25,156	00:22,250	00:23,959	2000000
	(Q4)	00:22,125	00:22,407	00:21,907	00:22,211	00:21,360	00:22,002	2000000
Casting	(Q5)	00:11,351	00:11,333	00:11,450	00:11,188	00:11,281	00:11,321	1574113
Expresiones regulares	(Q6)	00:07,854	00:07,809	00:07,781	00:08,094	00:07,720	00:07,852	981593
	(Q7)	00:15,390	00:13,749	00:13,844	00:13,564	00:13,609	00:14,031	1675642
Encontrar referencias	(Q8)	00:20,968	00:19,113	00:18,954	00:18,437	00:18,827	00:19,260	1980184
	(Q9)	00:01,893	00:01,109	00:01,010	00:01,016	00:01,109	00:01,227	15
Select anidado	(Q10)	00:13,311	00:15,359	00:13,156	00:16,406	00:13,922	00:14,431	295847
Joins	(Q11)	00:36,958	00:35,872	00:35,800	00:35,610	00:35,672	00:35,982	4000000
	(Q12)	00:22,063	00:21,250	00:21,782	00:21,157	00:20,751	00:21,401	1980184
	(Q13)	00:17,750	00:17,375	00:17,343	00:17,312	00:17,437	00:17,443	2000000
Búsqueda por índice	(Q14)	00:02,811	00:02,874	00:02,952	00:02,797	00:02,851	00:02,857	116733
Agregación	(Q15)	00:01,367	00:00,937	00:00,938	00:01,031	00:00,890	00:01,033	20089
	(Q16)	00:00,594	00:00,516	00:00,500	00:00,519	00:00,531	00:00,532	15
Inserción	(Q17)	00:00,037	00:00,031	00:00,030	00:00,016	00:00,063	00:00,035	100
Actualización	(Q18)	00:01,146	00:01,703	00:01,984	00:01,828	00:02,203	00:01,773	133408
	(Q19)	00:16,427	00:19,833	00:21,039	00:17,153	00:25,181	00:19,927	2000000
Eliminación	(Q20)	00:01,656	00:01,537	00:01,603	00:01,659	00:01,701	00:01,631	1692489
	(Q21)	00:01,622	00:01,538	00:01,752	00:01,609	00:01,595	00:01,623	2000000

Tabla 15: Matriz de toma de datos SQL Server 2016 en memoria.

(Elaboración Propia).

#### 4.1.4 SQL Server 2016 segunda ejecución sin liberar memoria – Evaluación 4.

SQL Server 2016					
Indicadores	N° Query	Sin Liberar Memoria			N° Registros
		En Disco	Mixtas	En Memoria	
Coincidencias exactas	(Q1)	00:02,379	-	00:02,218	16389
	(Q2)	00:01,224	00:01,156	00:09,966	36121
Acceso ordenado	(Q3)	00:34,343	-	00:22,001	2000000
	(Q4)	00:22,719	-	00:21,468	2000000
Casting	(Q5)	00:12,316	-	00:11,360	1574113
Expresiones regulares	(Q6)	00:08,129	-	00:07,672	981593
	(Q7)	00:16,688	00:17,688	00:15,141	1675642
Encontrar referencias	(Q8)	00:21,203	00:20,797	00:19,016	1980184
	(Q9)	00:01,421	00:01,422	00:01,110	15
Select anidado	(Q10)	00:05,891	00:17,468	00:14,634	295847
Joins	(Q11)	00:38,859	00:35,641	00:35,882	4000000
	(Q12)	00:24,172	00:21,516	00:21,032	1980184
	(Q13)	00:20,547	00:18,478	00:17,610	2000000
Búsqueda por índice	(Q14)	00:02,863	-	00:02,672	116733
Agregación	(Q15)	00:00,906	-	00:00,922	20089
	(Q16)	00:00,671	-	00:00,609	15
Inserción	(Q17)	00:00,280	-	00:00,016	100
Actualización	(Q18)	06:05,391	-	00:01,765	133408
	(Q19)	23:32,081	-	00:19,771	2000000
Eliminación	(Q20)	-	-	-	1692489
	(Q21)	-	-	-	2000000

Tabla 16: Matriz de toma de datos SQL Server 2016 sin liberar memoria.

(Elaboración Propia).

#### 4.1.5 SQL Server 2016 promedios – Evaluación 5.

SQL Server 2016				
Indicadores	N° Query	En Disco	Mixtas	En Memoria
Coincidencias exactas	(Q1)	00:30,378	-	00:02,418
	(Q2)	00:38,613	00:30,583	00:10,182
Acceso ordenado	(Q3)	02:14,356	-	00:23,959
	(Q4)	01:53,983	-	00:22,002
Casting	(Q5)	01:48,441	-	00:11,321
Expresiones regulares	(Q6)	00:14,659	-	00:07,852
	(Q7)	00:54,262	00:42,350	00:14,031
Encontrar referencias	(Q8)	01:00,153	00:50,976	00:19,260
	(Q9)	01:36,093	01:32,023	00:01,227
Select anidado	(Q10)	02:10,372	02:26,270	00:14,431
Joins	(Q11)	01:17,105	01:02,476	00:35,982
	(Q12)	00:57,141	00:46,279	00:21,401
	(Q13)	01:55,074	01:49,788	00:17,443
Búsqueda por índice	(Q14)	00:27,650	-	00:02,857
Agregación	(Q15)	00:09,662	-	00:01,033
	(Q16)	01:25,357	-	00:00,532
Inserción	(Q17)	00:00,333	-	00:00,035
Actualización	(Q18)	09:01,531	-	00:01,773
	(Q19)	26:18,021	-	00:19,927
Eliminación	(Q20)	01:35,737	-	00:01,631
	(Q21)	01:40,977	-	00:01,623

Tabla 17: Matriz de promedios SQL Server 2016.

(Elaboración Propia).

## 4.2 Pruebas de rendimiento Database Oracle 12c.

Para la toma de datos en Oracle 12c, al igual que en el caso anterior, se tomó en cuenta que los valores de tiempo indicados por el compilador coincidan con los valores tomados externamente y también, se manejaron 3 decimales para mayor exactitud en los cálculos.

### 4.2.1 Database Oracle 12c en disco – Evaluación 1.

Database Oracle 12c								
Indicadores	N° Query	En Disco						N° Registros
		1	2	3	4	5	P	
Coincidencias exactas	(Q1)	00:06,219	00:04,812	00:03,625	00:03,609	00:03,656	00:04,384	16389
	(Q2)	00:04,501	00:05,609	00:05,594	00:04,704	00:04,626	00:05,007	36121
Acceso ordenado	(Q3)	03:20,564	02:43,908	03:01,140	03:07,264	02:56,740	03:01,923	2000000
	(Q4)	02:34,312	02:45,044	02:42,228	02:25,472	03:10,208	02:43,453	2000000
Casting	(Q5)	00:37,600	00:35,181	00:36,393	00:35,906	00:35,906	00:36,197	1574113
Expresiones regulares	(Q6)	00:53,856	00:54,677	00:52,537	00:51,453	00:49,934	00:52,491	981593
	(Q7)	01:35,900	01:34,664	01:57,533	02:12,972	01:30,595	01:46,333	1675642
Encontrar referencias	(Q8)	02:43,298	02:25,125	02:02,435	02:57,800	02:58,678	02:37,467	1980184
	(Q9)	00:14,187	00:14,141	00:13,921	00:14,766	00:13,828	00:14,169	15
Select anidado	(Q10)	00:41,552	00:46,838	00:43,877	00:43,257	00:41,986	00:43,502	295847
Joins	(Q11)	07:02,880	06:08,440	06:27,800	05:37,480	05:21,240	06:07,568	4000000
	(Q12)	03:55,582	04:17,129	03:53,781	03:39,107	04:06,612	03:58,442	1980184
	(Q13)	03:34,241	03:58,840	04:09,360	03:45,180	03:40,500	03:49,624	2000000
Búsqueda por índice	(Q14)	00:06,954	00:07,598	00:07,426	00:07,365	00:07,568	00:07,382	116733
Agregación	(Q15)	00:00,797	00:00,750	00:00,721	00:00,719	00:00,720	00:00,741	20089
	(Q16)	00:12,062	00:11,109	00:10,875	00:10,844	00:11,094	00:11,197	15
Inserción	(Q17)	00:01,062	00:01,188	00:00,968	00:01,329	00:01,313	00:01,172	100
Actualización	(Q18)	04:30,564	03:49,422	04:09,423	03:53,923	04:23,798	04:09,426	133408
	(Q19)	03:54,111	04:44,640	03:25,485	02:59,466	03:34,789	03:43,698	2000000
Eliminación	(Q20)	03:52,922	04:03,467	03:49,481	03:40,898	03:58,133	03:52,980	1692489
	(Q21)	01:57,438	01:52,783	02:04,560	01:57,801	01:49,982	01:56,513	2000000

Tabla 18: Matriz de toma de datos Database Oracle 12c en disco.

(Elaboración Propia).

#### 4.2.2 Database Oracle 12c mixta – Evaluación 2.

Database Oracle 12c								
Indicadores	N° Query	Mixtas						N° Registros
		1	2	3	4	5	P	
Coincidencias exactas	(Q1)	-						16389
	(Q2)	00:05,672	00:04,578	00:05,860	00:04,938	00:05,031	00:05,216	36121
Acceso ordenado	(Q3)	-						2000000
	(Q4)	-						2000000
Casting	(Q5)	-						1574113
Expresiones regulares	(Q6)	-						981593
	(Q7)	01:13,963	01:20,112	01:24,829	01:19,987	01:10,829	01:17,944	1675642
Encontrar referencias	(Q8)	02:30,896	02:33,048	02:30,217	02:28,000	02:34,078	02:31,248	1980184
	(Q9)	00:14,501	00:15,188	00:14,202	00:14,688	00:14,266	00:14,569	15
Select anidado	(Q10)	00:34,719	00:30,355	00:29,627	00:30,012	00:33,921	00:31,727	295847
Joins	(Q11)	05:46,760	06:10,880	05:58,360	06:24,561	06:15,880	06:07,288	4000000
	(Q12)	04:04,117	04:07,444	03:55,127	04:08,533	03:48,256	04:00,695	1980184
	(Q13)	03:24,080	03:48,129	03:27,820	03:37,200	03:28,819	03:34,557	2000000
Búsqueda por índice	(Q14)	-						116733
Agregación	(Q15)	-						20089
	(Q16)	-						15
Inserción	(Q17)	-						100
Actualización	(Q18)	-						133408
	(Q19)	-						2000000
Eliminación	(Q20)	-						1692489
	(Q21)	-						2000000

Tabla 19: Matriz de toma de datos Database Oracle 12c mixta.

(Elaboración Propia).

#### 4.2.3 Database Oracle 12c en memoria – Evaluación 3.

Database Oracle 12c								
Indicadores	N° Query	En Memoria						N° Registros
		1	2	3	4	5	P	
Coincidencias exactas	(Q1)	00:00,765	00:00,343	00:00,296	00:00,571	00:00,266	00:00,448	16389
	(Q2)	00:00,719	00:00,656	00:00,657	00:00,625	00:00,624	00:00,656	36121
Acceso ordenado	(Q3)	01:04,252	01:04,688	00:58,480	01:02,500	01:05,436	01:03,071	2000000
	(Q4)	01:04,816	01:00,688	00:59,688	01:00,752	01:05,253	01:02,239	2000000
Casting	(Q5)	00:27,301	00:26,269	00:26,545	00:28,583	00:26,757	00:27,091	1574113
Expresiones regulares	(Q6)	00:21,442	00:21,809	00:21,102	00:21,454	00:21,935	00:21,548	981593
	(Q7)	00:46,553	00:43,932	00:42,152	00:42,260	00:41,576	00:43,295	1675642
Encontrar referencias	(Q8)	01:16,669	01:08,815	01:28,178	01:06,645	01:20,134	01:16,088	1980184
	(Q9)	00:01,550	00:03,218	00:02,969	00:02,968	00:02,953	00:02,732	15
Select anidado	(Q10)	00:06,656	00:06,266	00:05,516	00:05,515	00:06,392	00:06,069	295847
Joins	(Q11)	02:17,872	02:34,240	02:02,880	02:20,194	02:14,816	02:18,000	4000000
	(Q12)	01:05,714	01:13,275	01:07,944	01:07,572	01:08,205	01:08,542	1980184
	(Q13)	00:58,248	00:57,752	00:56,753	01:11,872	00:59,368	01:00,799	2000000
Búsqueda por índice	(Q14)	00:01,751	00:01,734	00:01,828	00:01,774	00:01,756	00:01,769	116733
Agregación	(Q15)	00:00,531	00:00,547	00:00,546	00:00,516	00:00,533	00:00,535	20089
	(Q16)	00:00,108	00:00,111	00:00,093	00:00,117	00:00,094	00:00,105	15
Inserción	(Q17)	00:01,624	00:01,062	00:00,983	00:00,959	00:00,909	00:01,107	100
Actualización	(Q18)	05:56,641	03:52,579	05:17,548	05:43,581	05:31,198	05:16,309	133408
	(Q19)	04:19,016	04:40,735	04:12,002	03:52,315	04:22,803	04:17,374	2000000
Eliminación	(Q20)	01:18,611	01:27,499	01:22,254	01:28,003	01:20,139	01:23,301	1692489
	(Q21)	02:27,126	02:12,828	02:19,563	02:16,824	02:26,350	02:20,538	2000000

Tabla 20: Matriz de toma de datos Database Oracle 12c en memoria.

(Elaboración Propia).

#### 4.2.4 Database Oracle 12c segunda ejecución sin liberar memoria – Evaluación 4.

Database Oracle 12c					
Indicadores	N° Query	Sin Liberar Memoria			N° Registros
		En Disco	Mixtas	En Memoria	
Coincidencias exactas	(Q1)	00:03,547	-	00:00,328	16389
	(Q2)	00:04,766	00:05,046	00:00,608	36121
Acceso ordenado	(Q3)	03:27,644	-	01:06,312	2000000
	(Q4)	02:48,276	-	00:59,708	2000000
Casting	(Q5)	00:36,645	-	00:25,976	1574113
Expresiones regulares	(Q6)	00:49,588	-	00:21,412	981593
	(Q7)	01:38,682	01:36,609	00:41,261	1675642
Encontrar referencias	(Q8)	02:35,398	02:36,801	01:08,000	1980184
	(Q9)	00:14,030	00:14,922	00:02,751	15
Select anidado	(Q10)	00:42,243	00:34,035	00:06,204	295847
Joins	(Q11)	04:45,640	05:45,561	02:10,752	4000000
	(Q12)	03:55,226	03:55,761	01:06,708	1980184
	(Q13)	03:22,740	03:43,800	00:57,248	2000000
Búsqueda por índice	(Q14)	00:07,586	-	00:01,766	116733
Agregación	(Q15)	00:00,735	-	00:00,594	20089
	(Q16)	00:11,453	-	00:00,125	15
Inserción	(Q17)	00:01,417	-	00:00,971	100
Actualización	(Q18)	03:25,078	-	05:56,657	133408
	(Q19)	03:07,529	-	04:05,165	2000000
Eliminación	(Q20)	-	-	-	1692489
	(Q21)	-	-	-	2000000

Tabla 21: Matriz de toma de datos Database Oracle 12c sin liberar memoria.

(Elaboración Propia).

#### 4.2.5 Database Oracle 12c promedios – Evaluación 5.

Database Oracle 12c				
Indicadores	N° Query	En Disco	Mixtas	En Memoria
Coincidencias exactas	(Q1)	00:04,384	-	00:00,448
	(Q2)	00:05,007	00:05,216	00:00,656
Acceso ordenado	(Q3)	03:01,923	-	01:03,071
	(Q4)	02:43,453	-	01:02,239
Casting	(Q5)	00:36,197	-	00:27,091
Expresiones regulares	(Q6)	00:52,491	-	00:21,548
	(Q7)	01:46,333	01:17,944	00:43,295
Encontrar referencias	(Q8)	02:37,467	02:31,248	01:16,088
	(Q9)	00:14,169	00:14,569	00:02,732
Select anidado	(Q10)	00:43,502	00:31,727	00:06,069
Joins	(Q11)	06:07,568	06:07,288	02:18,000
	(Q12)	03:58,442	04:00,695	01:08,542
	(Q13)	03:49,624	03:34,557	01:00,799
Búsqueda por índice	(Q14)	00:07,382	-	00:01,769
Agregación	(Q15)	00:00,741	-	00:00,535
	(Q16)	00:11,197	-	00:00,105
Inserción	(Q17)	00:01,172	-	00:01,107
Actualización	(Q18)	04:09,426	-	05:16,309
	(Q19)	03:43,698	-	04:17,374
Eliminación	(Q20)	03:52,980	-	01:23,301
	(Q21)	01:56,513	-	02:20,538

Tabla 22: Matriz de promedios Database Oracle 12c.

(Elaboración Propia).

### 4.3 Pruebas de rendimiento Oracle TimesTen.

En el caso de TimesTen al ser una base de datos en memoria se realizó solo la toma de datos en memoria. En cuanto al procedimiento de toma de datos se siguió los mismos estándares de los dos casos anteriores.

#### 4.3.1 Oracle TimesTen en memoria – Evaluación 3.

Times Ten								
Indicadores	N° Query	En Memoria						N° Registros
		1	2	3	4	5	P	
Coincidencias exactas	(Q1)	00:01,562	00:01,765	00:01,625	00:01,673	00:01,531	00:01,631	16389
	(Q2)	00:03,731	00:03,875	00:03,719	00:03,844	00:03,766	00:03,787	36121
Acceso ordenado	(Q3)	02:42,060	02:41,316	02:44,188	02:43,436	02:39,188	02:42,038	2000000
	(Q4)	02:45,624	02:39,876	02:41,004	02:39,576	02:41,936	02:41,603	2000000
Casting	(Q5)	01:30,609	01:29,973	01:31,100	01:30,216	01:27,659	01:29,911	1574113
Expresiones regulares	(Q6)	01:16,605	01:15,918	01:15,488	01:16,013	01:16,101	01:16,025	981593
	(Q7)	02:03,213	01:56,095	01:59,336	01:59,074	01:58,615	01:59,267	1675642
Encontrar referencias	(Q8)	02:03,021	02:01,100	02:00,855	02:00,728	02:02,403	02:01,621	1980184
	(Q9)	00:02,078	00:01,985	00:02,141	00:01,969	00:01,938	00:02,022	15
Select anidado	(Q10)	00:26,781	00:26,438	00:26,562	00:26,376	00:26,501	00:26,532	295847
Joins	(Q11)	00:50,141	00:49,297	00:50,062	00:51,265	00:49,265	00:50,006	4000000
	(Q12)	03:20,046	03:12,569	03:07,995	03:09,915	03:10,110	03:12,127	1980184
	(Q13)	02:42,064	02:41,564	02:40,628	02:40,004	02:41,316	02:41,115	2000000
Búsqueda por índice	(Q14)	00:06,562	00:06,327	00:06,454	00:06,391	00:06,453	00:06,437	116733
Agregación	(Q15)	00:01,797	00:01,609	00:01,641	00:01,625	00:01,578	00:01,650	20089
	(Q16)	00:00,390	00:00,359	00:00,454	00:00,392	00:00,344	00:00,388	15
Inserción	(Q17)	00:02,563	00:02,313	00:02,547	00:02,437	00:02,328	00:02,438	100
Actualización	(Q18)	00:00,875	00:00,751	00:00,844	00:00,812	00:00,843	00:00,825	133408
	(Q19)	00:05,141	00:10,391	00:11,843	00:09,375	00:08,781	00:09,106	2000000
Eliminación	(Q20)	00:49,875	00:50,011	00:49,937	00:50,163	00:49,732	00:49,944	1692489
	(Q21)	00:38,172	00:38,765	00:37,985	00:38,201	00:38,406	00:38,306	2000000

Tabla 23: Matriz de toma de datos TimesTen en memoria.

(Elaboración Propia).

#### 4.3.2 Oracle TimesTen segunda ejecución sin liberar memoria – Evaluación 4.

Times Ten			
Indicadores	N° Query	En Memoria	N° Registros
Coincidencias exactas	(Q1)	00:01,640	16389
	(Q2)	00:03,735	36121
Acceso ordenado	(Q3)	02:42,004	2000000
	(Q4)	02:40,376	2000000
Casting	(Q5)	01:31,302	1574113
Expresiones regulares	(Q6)	01:14,784	981593
	(Q7)	01:56,822	1675642
Encontrar referencias	(Q8)	02:00,360	1980184
	(Q9)	00:02,047	15
Select anidado	(Q10)	00:26,563	295847
Joins	(Q11)	00:49,406	4000000
	(Q12)	03:10,220	1980184
	(Q13)	02:38,372	2000000
Búsqueda por índice	(Q14)	00:06,438	116733
Agregación	(Q15)	00:01,594	20089
	(Q16)	00:00,375	15
Inserción	(Q17)	00:02,374	100
Actualización	(Q18)	00:00,859	133408
	(Q19)	00:11,140	2000000
Eliminación	(Q20)	-	-
	(Q21)	-	-

Tabla 24: Matriz de toma de datos TimesTen sin liberar memoria.

(Elaboración Propia).

#### 4.4 Resultados Finales.

##### 4.4.1 SQL Server 2016.

##### 4.4.1.1 Promedios Finales.

Para poder mostrar gráficamente los resultados de los promedios obtenidos, se agrupó los datos de forma que los tiempos resultantes se puedan apreciar de una mejor forma, primero las consultas que se pueden realizar en forma de en disco, mixta y en memoria.

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

Indicadores	N° Query	En Disco	Mixtas	En Memoria
Coincidencias exactas	(Q2)	00:38,613	00:30,583	00:10,182
Expresiones regulares	(Q7)	00:54,262	00:42,350	00:14,031
Encontrar referencias	(Q8)	01:00,153	00:50,976	00:19,260
	(Q9)	01:36,093	01:32,023	00:01,227
Select anidado	(Q10)	02:10,372	02:26,270	00:14,431
Joins	(Q11)	01:17,105	01:02,476	00:35,982
	(Q12)	00:57,141	00:46,279	00:21,401
	(Q13)	01:55,074	01:49,788	00:17,443

Tabla 25: Extracto 1 En Disco vs Mixta vs En Memoria - Matriz de promedios SQL Server 2016.

(Elaboración Propia).

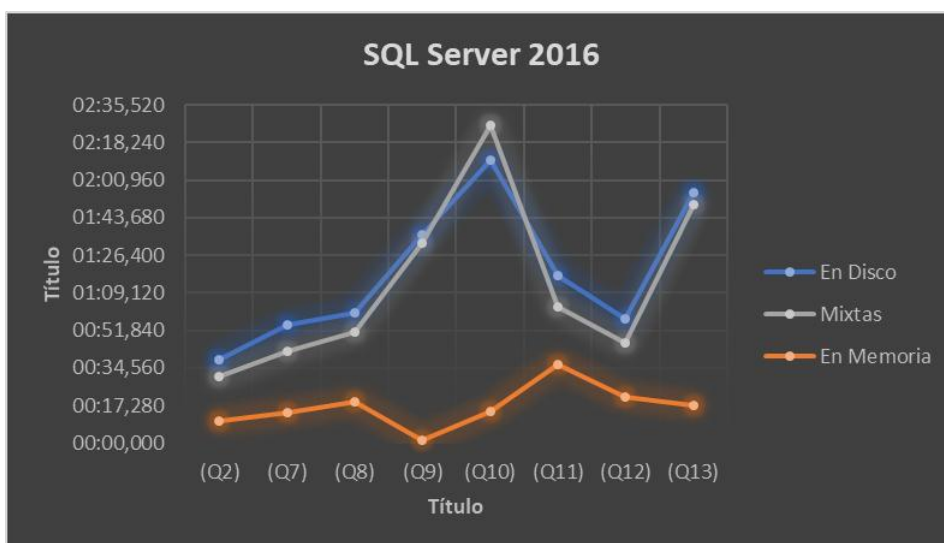


Gráfico 16: Grafico E1 - En Disco vs Mixta vs En Memoria SQL Server 2016.

(Elaboración Propia).

Segundo, dado que no todas las consultas se pueden realizar de forma mixta, en el siguiente extracto y gráfico se muestra las consultas correspondientes a disco y en memoria.

Indicadores	N° Query	En Disco	En Memoria
Coincidencias exactas	(Q1)	00:30,378	00:02,418
Acceso ordenado	(Q3)	02:14,356	00:23,959
	(Q4)	01:53,983	00:22,002
Casting	(Q5)	01:48,441	00:11,321
Expresiones regulares	(Q6)	00:14,659	00:07,852
Búsqueda por índice	(Q14)	00:27,650	00:02,857
Agregación	(Q15)	00:09,662	00:01,033
	(Q16)	01:25,357	00:00,532
Inserción	(Q17)	00:00,333	00:00,035

Tabla 26: Extracto 2 En Disco vs En Memoria - Matriz de promedios SQL Server 2016.

(Elaboración Propia).

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

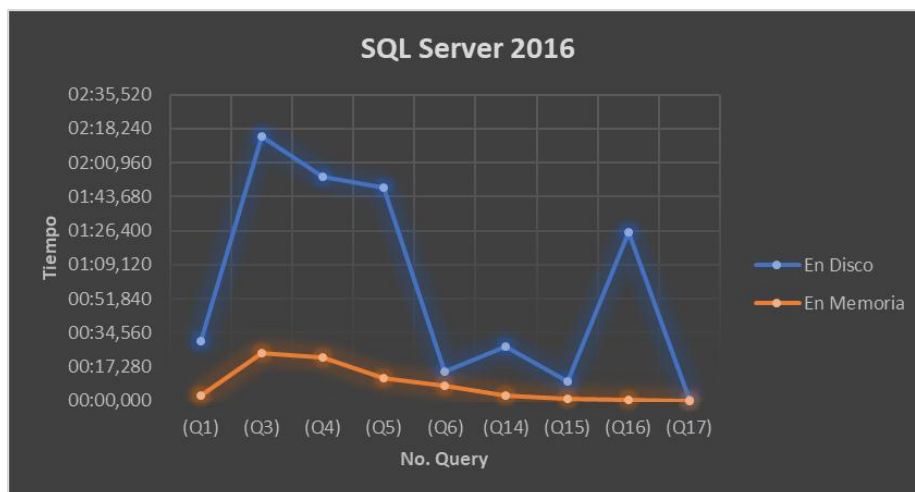


Gráfico 17: Gráfico E2 - En Disco vs En Memoria SQL Server 2016.

(Elaboración Propia).

Tercero, las consultas de actualización y eliminación, que no se pueden realizar de forma mixta y no serían muy bien visualizadas en el segundo grupo, se han dividido en un tercer grupo para su correcta visualización.

Indicadores	N° Query	En Disco	En Memoria
Actualización	(Q18)	09:01,531	00:01,773
	(Q19)	26:18,021	00:19,927
Eliminación	(Q20)	01:35,737	00:01,631
	(Q21)	01:40,977	00:01,623

Tabla 27: Extracto 3 En Disco vs En Memoria - Matriz de promedios SQL Server 2016.

(Elaboración Propia).

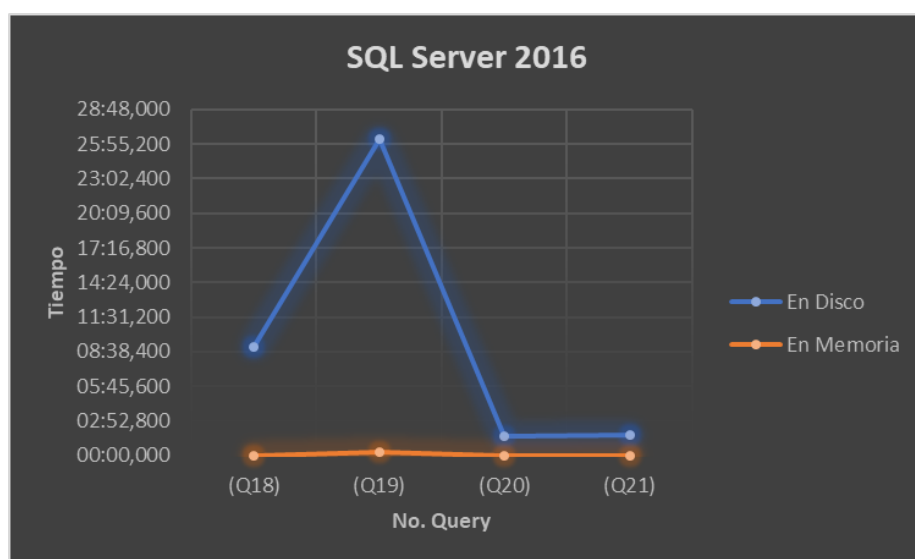


Gráfico 18: Gráfico E3 - En Disco vs En Memoria SQL Server 2016.

(Elaboración Propia).

#### 4.4.1.2 Matriz de Calificación.

Para la calificación en cuanto a los resultados obtenidos se mantuvo los siguientes criterios:

- Si la consulta cuenta con los tres tipos en Disco, Mixta y en Memoria se califica al menor tiempo con 3 puntos, seguido del siguiente tiempo con 2 puntos y el mayor de los tres tiempos con solo 1 punto.
- Si la consulta cuenta con solo datos de tipos en Disco y en Memoria se asigna al menor tiempo 3 puntos y 1.5 puntos al siguiente.
- Para el resultado final, se divide la suma total de los resultados en cada consulta para el numero de participaciones que cuenta cada tipo, si es el caso de Disco 21, Mixtas 8 y en Memoria 21.

SQL Server 2016				
Indicadores	N° Query	En Disco	Mixtas	En Memoria
Coincidencias exactas	(Q1)	1,5	-	3
	(Q2)	1	2	3
Acceso ordenado	(Q3)	1,5	-	3
	(Q4)	1,5	-	3
Casting	(Q5)	1,5	-	3
Expresiones regulares	(Q6)	1,5	-	3
	(Q7)	1	2	3
Encontrar referencias	(Q8)	1	2	3
	(Q9)	1	2	3
Select anidado	(Q10)	2	1	3
Joins	(Q11)	1	2	3
	(Q12)	1	2	3
	(Q13)	1	2	3
Búsqueda por índice	(Q14)	1,5	-	3
Agregación	(Q15)	1,5	-	3
	(Q16)	1,5	-	3
Inserción	(Q17)	1,5	-	3
Actualización	(Q18)	1,5	-	3
	(Q19)	1,5	-	3
Eliminación	(Q20)	1,5	-	3
	(Q21)	1,5	-	3
Puntaje Final		1,36	1,88	3

Tabla 28: Matriz de calificación SQL Server 2016.

(Elaboración Propia).

#### 4.4.1.3 Promedios Finales Segunda ejecución.

De igual forma que en el caso anterior, vamos a dividir en tres grupos los cuales permitirán una mejor visualización de los datos.

Primer grupo conformado por los datos que pueden ser tomados en los tres tipos: en Disco, Mixtas y en Memoria.

Indicadores	N° Query	En Disco	Mixtas	En Memoria
Coincidencias exactas	(Q2)	00:01,224	00:01,156	00:09,966
Expresiones regulares	(Q7)	00:16,688	00:17,688	00:15,141
Encontrar referencias	(Q8)	00:21,203	00:20,797	00:19,016
	(Q9)	00:01,421	00:01,422	00:01,110
Select anidado	(Q10)	00:05,891	00:17,468	00:14,634
Joins	(Q11)	00:38,859	00:35,641	00:35,882
	(Q12)	00:24,172	00:21,516	00:21,032
	(Q13)	00:20,547	00:18,478	00:17,610

Tabla 29: Extracto 1 En Disco vs Mixta vs En Memoria - Matriz de toma de datos SQL Server 2016 sin liberar memoria.

(Elaboración Propia).

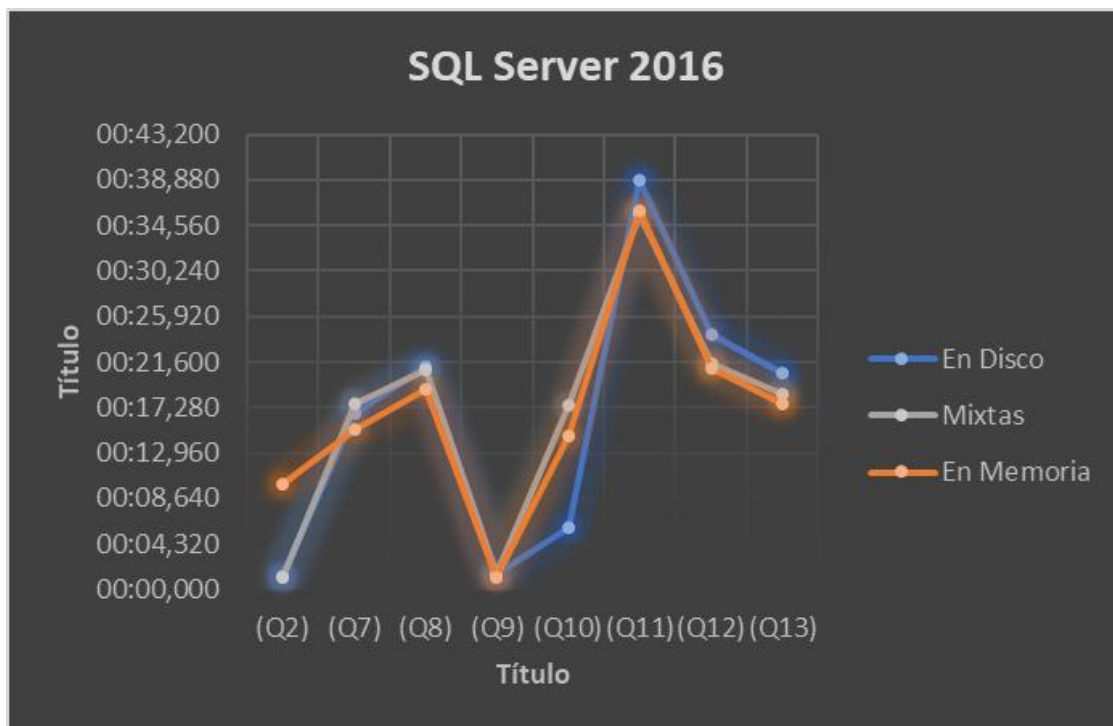


Gráfico 19: Grafico E1 – En Disco vs Mixta vs En Memoria SQL Server 2016 sin liberar memoria.

(Elaboración Propia).

Segundo grupo conformado por los datos que son viables en Disco y Memoria.

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

Indicadores	N° Query	En Disco	En Memoria
Coincidencias exactas	(Q1)	00:02,379	00:02,218
Acceso ordenado	(Q3)	00:34,343	00:22,001
	(Q4)	00:22,719	00:21,468
Casting	(Q5)	00:12,316	00:11,360
Expresiones regulares	(Q6)	00:08,129	00:07,672
Búsqueda por índice	(Q14)	00:02,863	00:02,672
Agregación	(Q15)	00:00,906	00:00,922
	(Q16)	00:00,671	00:00,609
Inserción	(Q17)	00:00,280	00:00,016

Tabla 30: Extracto 2 En Disco vs En Memoria - Matriz de promedios SQL Server 2016 sin liberar memoria.

(Elaboración Propia).

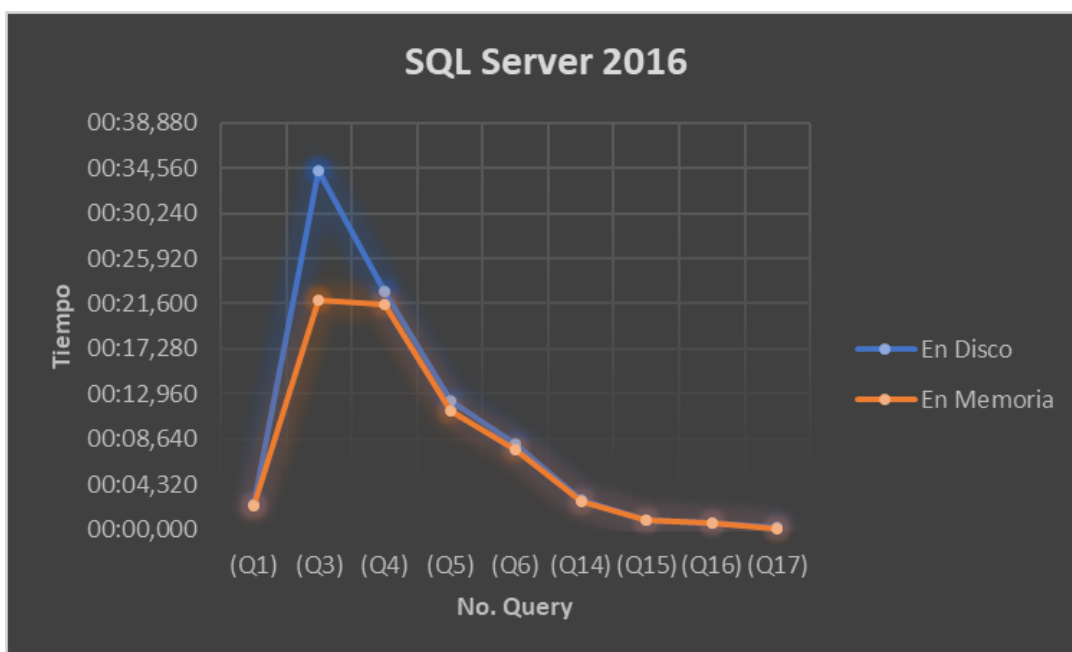


Gráfico 20: Gráfico E2 - En Disco vs En Memoria SQL Server 2016 sin liberar memoria.

(Elaboración Propia).

Tercer grupo, los datos que son viables en Disco y Memoria y que además se diferencian mucho en tiempo en cuanto a los del segundo grupo.

Indicadores	N° Query	En Disco	En Memoria
Actualización	(Q18)	06:05,391	00:01,765
	(Q19)	23:32,081	00:19,771

Tabla 31: Extracto 3 En Disco vs En Memoria - Matriz de promedios SQL Server 2016 sin liberar memoria.

(Elaboración Propia).

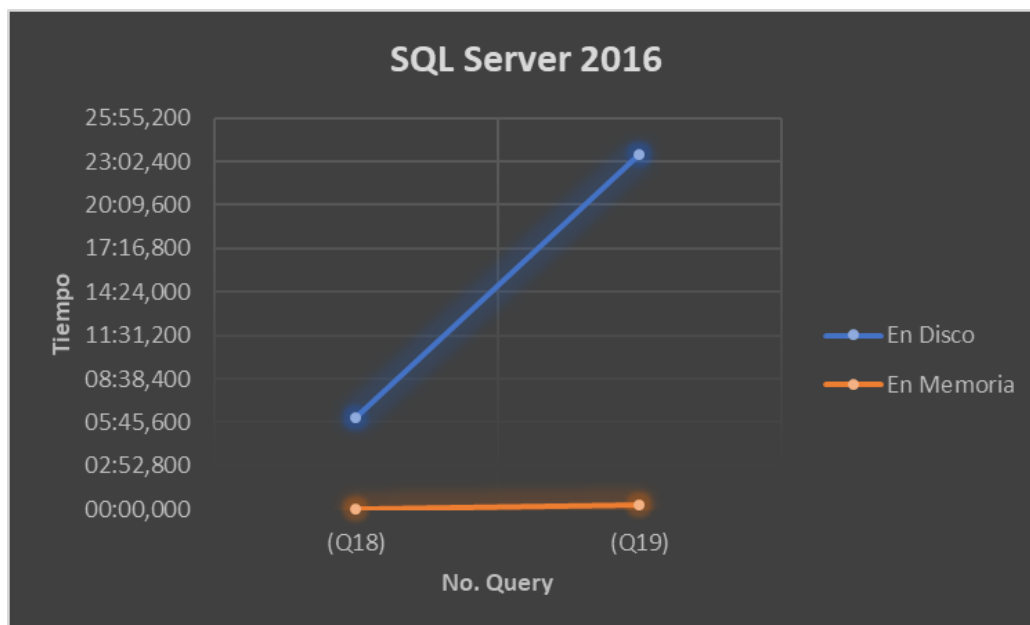


Gráfico 21: Gráfico E3 – En Disco vs En Memoria SQL Server 2016 sin liberar memoria.

(Elaboración Propia).

Nota: Las Eliminaciones no fueron tomadas en cuenta en este caso sin liberar memoria, ya que no se puede eliminar el mismo elemento en dos ocasiones seguidas.

#### 4.4.1.4 Contraste Primeras ejecuciones vs Segundas ejecuciones sin liberar memoria.

Esta comparativa nos muestra los resultados de cómo opera el gestor de bases de datos SQL Server luego de ejecutar por primera vez una consulta y posteriormente ejecutar la misma consulta.

Primero podemos observar la comparación entre las primeras ejecuciones de disco y las segundas ejecuciones en disco sin liberar memoria.

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

Indicadores	N° Query	2da En Disco	En Disco
Coincidencias exactas	(Q1)	00:02,379	00:30,378
	(Q2)	00:01,224	00:38,613
Acceso ordenado	(Q3)	00:34,343	02:14,356
	(Q4)	00:22,719	01:53,983
Casting	(Q5)	00:12,316	01:48,441
Expresiones regulares	(Q6)	00:08,129	00:14,659
	(Q7)	00:16,688	00:54,262
Encontrar referencias	(Q8)	00:21,203	01:00,153
	(Q9)	00:01,421	01:36,093
Select anidado	(Q10)	00:05,891	02:10,372
Joins	(Q11)	00:38,859	01:17,105
	(Q12)	00:24,172	00:57,141
	(Q13)	00:20,547	01:55,074
Búsqueda por índice	(Q14)	00:02,863	00:27,650
Agregación	(Q15)	00:00,906	00:09,662
	(Q16)	00:00,671	01:25,357
Inserción	(Q17)	00:00,280	00:00,333
Actualización	(Q18)	06:05,391	09:01,531
	(Q19)	23:32,081	26:18,021

Tabla 32: Matriz Primera ejecución en Disco vs Segunda ejecución en Disco sin liberar memoria SQL Server 2016.

(Elaboración Propia).

Para que se pueda observar de mejor forma el grafico se omitió las consultas 18 y 19 de actualización ya que estas tienen muchísima diferencia de tiempos en el gráfico, pero hay que señalar que al igual que las otras consultas, las consultas de actualización son de menor tiempo en la segunda ejecución sin liberar memoria.



Gráfico 22: Gráfico Primera ejecución en Disco vs Segunda ejecución en Disco sin liberar memoria SQL Server 2016.

(Elaboración Propia).

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

Comparación entre las primeras ejecuciones mixtas y las segundas ejecuciones mixtas sin liberar memoria.

Indicadores	Nº Query	Mixta	2da Mixta
Coincidencias exactas	(Q2)	00:30,583	00:01,156
Expresiones regulares	(Q7)	00:42,350	00:17,688
Encontrar referencias	(Q8)	00:50,976	00:20,797
	(Q9)	01:32,023	00:01,422
Select anidado	(Q10)	02:26,270	00:17,468
Joins	(Q11)	01:02,476	00:35,641
	(Q12)	00:46,279	00:21,516
	(Q13)	01:49,788	00:18,478

Tabla 33: Matriz Primera ejecución Mixta vs Segunda ejecución Mixta sin liberar memoria SQL Server 2016.

(Elaboración Propia).

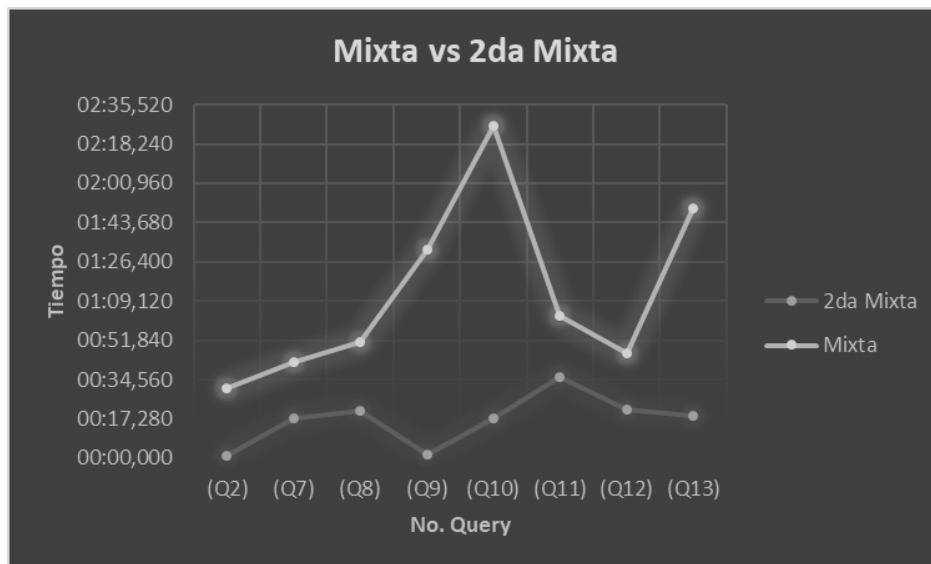


Gráfico 23: Gráfico Primera ejecución Mixta vs Segunda ejecución Mixta sin liberar memoria SQL Server 2016.

(Elaboración Propia).

Comparación entre las primeras ejecuciones en memoria y las segundas ejecuciones en memoria.

Indicadores	N° Query	En Memoria	2da En Memoria
Coincidencias exactas	(Q1)	00:02,418	00:02,218
	(Q2)	00:10,182	00:09,966
Acceso ordenado	(Q3)	00:23,959	00:22,001
	(Q4)	00:22,002	00:21,468
Casting	(Q5)	00:11,321	00:11,360
Expresiones regulares	(Q6)	00:07,852	00:07,672
	(Q7)	00:14,031	00:15,141
Encontrar referencias	(Q8)	00:19,260	00:19,016
	(Q9)	00:01,227	00:01,110
Select anidado	(Q10)	00:14,431	00:14,634
Joins	(Q11)	00:35,982	00:35,882
	(Q12)	00:21,401	00:21,032
	(Q13)	00:17,443	00:17,610
Búsqueda por índice	(Q14)	00:02,857	00:02,672
Agregación	(Q15)	00:01,033	00:00,922
	(Q16)	00:00,532	00:00,609
Inserción	(Q17)	00:00,035	00:00,016
Actualización	(Q18)	00:01,773	00:01,765
	(Q19)	00:19,927	00:19,771

Tabla 34: Matriz Primera ejecución en Memoria vs Segunda ejecución en Memoria SQL Server 2016.

(Elaboración Propia).

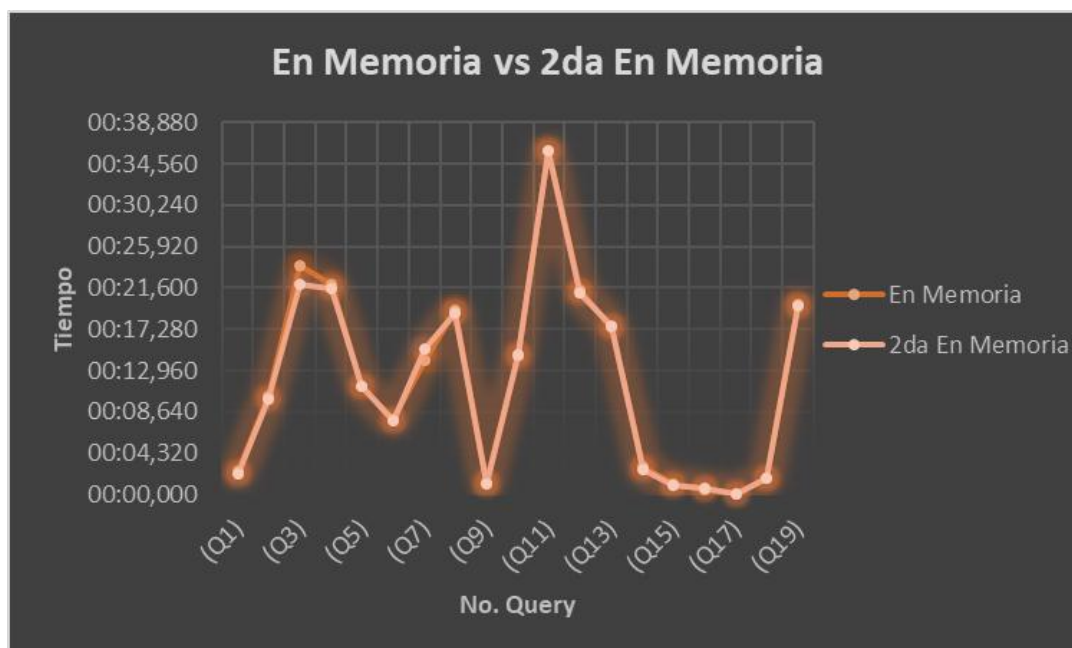


Gráfico 24: Grafico Primera ejecución en Memoria vs Segunda ejecución en Memoria SQL Server 2016.

(Elaboración Propia).

Y, por último, es lógico comparar la segunda ejecución en disco vs la forma en memoria ya que este compilador guarda esa información en memoria y hace que mientras estamos en disco también se pueda operar en memoria siempre y cuando la consulta se repita o en algunos casos se use la información de las mismas tablas de origen.

Indicadores	N° Query	2da En Disco	En Memoria
Coincidencias exactas	(Q1)	00:02,379	00:02,418
	(Q2)	00:01,224	00:10,182
Acceso ordenado	(Q3)	00:34,343	00:23,959
	(Q4)	00:22,719	00:22,002
Casting	(Q5)	00:12,316	00:11,321
Expresiones regulares	(Q6)	00:08,129	00:07,852
	(Q7)	00:16,688	00:14,031
Encontrar referencias	(Q8)	00:21,203	00:19,260
	(Q9)	00:01,421	00:01,227
Select anidado	(Q10)	00:05,891	00:14,431
Joins	(Q11)	00:38,859	00:35,982
	(Q12)	00:24,172	00:21,401
	(Q13)	00:20,547	00:17,443
Búsqueda por índice	(Q14)	00:02,863	00:02,857
Agregación	(Q15)	00:00,906	00:01,033
	(Q16)	00:00,671	00:00,532
Inserción	(Q17)	00:00,280	00:00,035
Actualización	(Q18)	06:05,391	00:01,773
	(Q19)	23:32,081	00:19,927

Tabla 35: Matriz Primera ejecución en Memoria vs Segunda ejecución en Disco SQL Server 2016.

(Elaboración Propia).

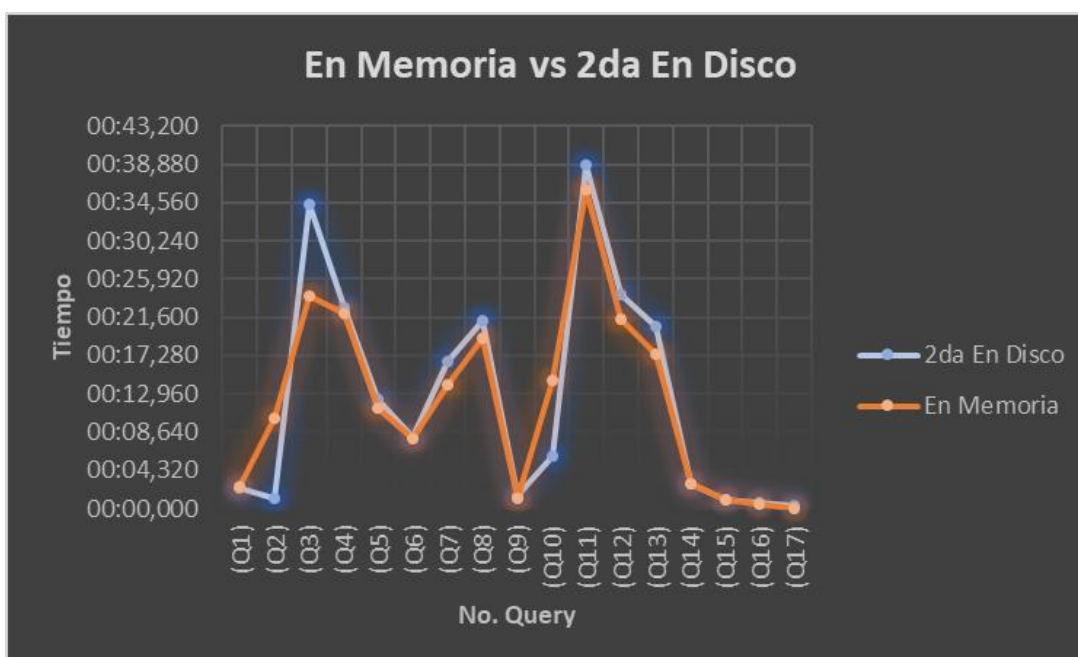


Gráfico 25: Gráfico Primera ejecución en Memoria vs Segunda ejecución en Disco SQL Server 2016.

(Elaboración Propia).

#### 4.4.2 Oracle 12c.

##### 4.4.2.1 Promedios Finales.

De la misma manera en que se procedió con SQL Server 2016, para poder apreciar los resultados de los promedios obtenidos se agrupó los datos de forma que se puedan apreciar los tiempos resultantes, primero las consultas que se pueden realizar en disco, mixta y memoria.

Indicadores	N° Query	En Disco	Mixtas	En Memoria
Coincidencias exactas	(Q2)	00:05,007	00:05,216	00:00,656
Expresiones regulares	(Q7)	01:46,333	01:17,944	00:43,295
Encontrar referencias	(Q8)	02:37,467	02:31,248	01:16,088
	(Q9)	00:14,169	00:14,569	00:02,732
Select anidado	(Q10)	00:43,502	00:31,727	00:06,069
Joins	(Q11)	06:07,568	06:07,288	02:18,000
	(Q12)	03:58,442	04:00,695	01:08,542
	(Q13)	03:49,624	03:34,557	01:00,799

Tabla 36: Extracto 1 En Disco vs Mixta vs En Memoria - Matriz de promedios Database Oracle 12c.

(Elaboración Propia).

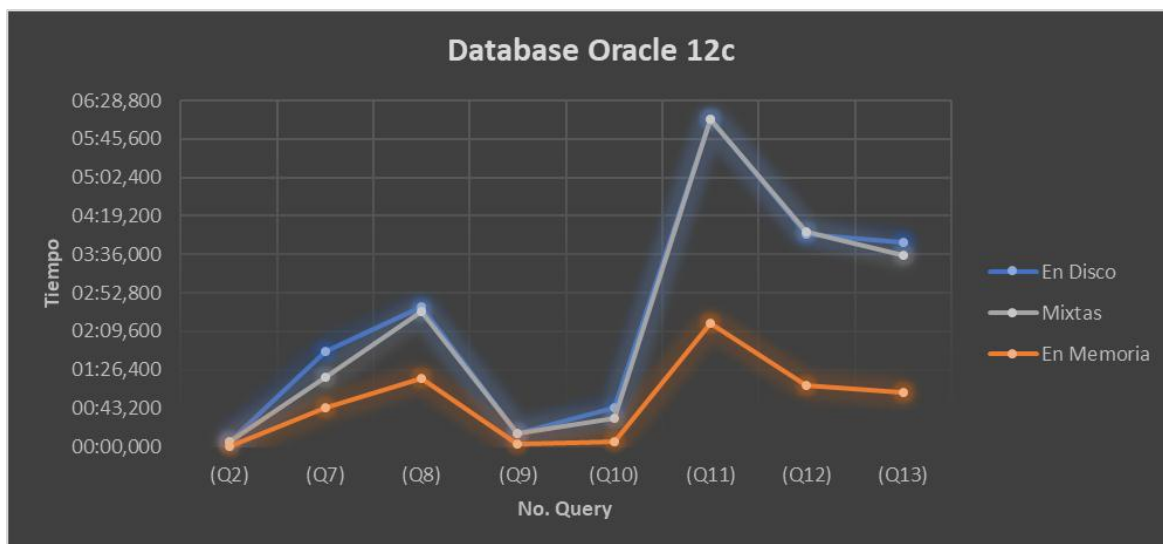


Gráfico 26: Grafico E1 - En Disco vs Mixta vs En Memoria Database Oracle 12c.

(Elaboración Propia).

Segundo, dado que no todas las consultas se pueden realizar de forma mixta, en el siguiente extracto y gráfico se muestran las consultas correspondientes a este caso.

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

Indicadores	N° Query	En Disco	En Memoria
Coincidencias exactas	(Q1)	00:04,384	00:00,448
Acceso ordenado	(Q3)	03:01,923	01:03,071
	(Q4)	02:43,453	01:02,239
Casting	(Q5)	00:36,197	00:27,091
Expresiones regulares	(Q6)	00:52,491	00:21,548
Búsqueda por índice	(Q14)	00:07,382	00:01,769
Agregación	(Q15)	00:00,741	00:00,535
	(Q16)	00:11,197	00:00,105
Inserción	(Q17)	00:01,172	00:01,107

Tabla 37: Extracto 2 En Disco vs En Memoria - Matriz de promedios Database Oracle 12c.

(Elaboración Propia).

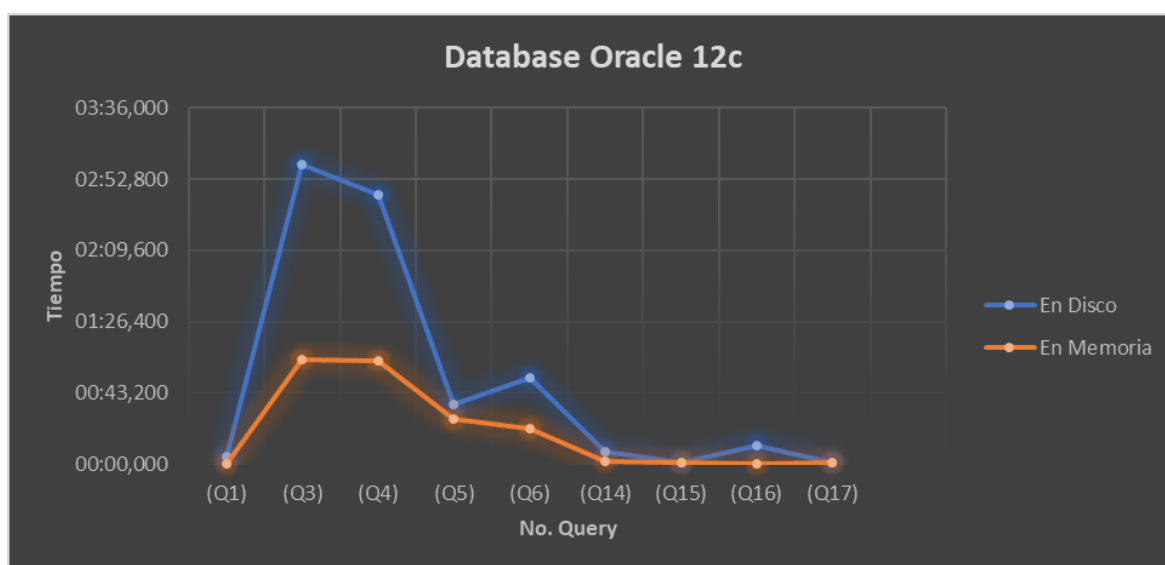


Gráfico 27: Gráfico E2 - En Disco vs En Memoria Database Oracle 12c.

(Elaboración Propia).

Tercero, las consultas de actualización y eliminación, que no se pueden realizar de forma mixta y no serían muy bien visualizadas en el segundo grupo, se han dividido en un tercer grupo para su correcta visualización.

Indicadores	N° Query	En Disco	En Memoria
Actualización	(Q18)	04:09,426	05:16,309
	(Q19)	03:43,698	04:17,374
Eliminación	(Q20)	03:52,980	01:23,301
	(Q21)	01:56,513	02:20,538

Tabla 38: Extracto 3 En Disco vs En Memoria - Matriz de promedios Database Oracle 12c.

(Elaboración Propia).

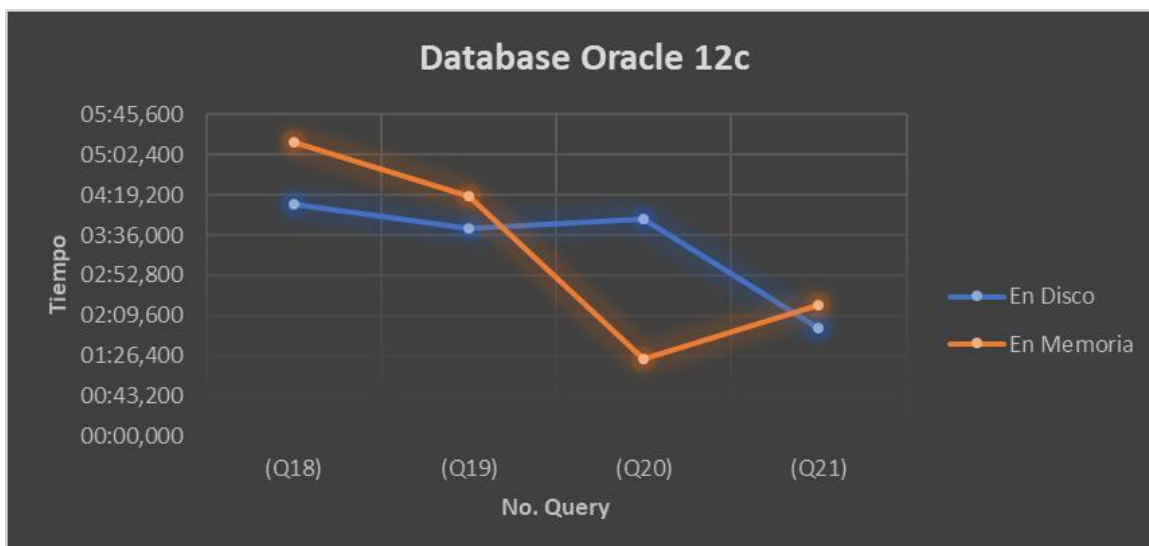


Gráfico 28: Grafico E3 - En Disco vs En Memoria Database Oracle 12c.

(Elaboración Propia).

#### 4.4.2.2 Matriz de Calificación.

De la misma forma que con la matriz de calificación de SQL Server para la calificación en cuanto a los resultados obtenidos se mantuvo los mismos criterios:

- Si la consulta cuenta con los tres tipos en Disco, Mixta y en Memoria se califica al menor tiempo con 3 puntos, seguido del siguiente tiempo con 2 puntos y el mayor de los tres tiempos con solo 1 punto.
- Si la consulta cuenta con solo datos de tipos en Disco y en Memoria se asigna al menor tiempo 3 puntos y 1.5 puntos al siguiente.
- Para el resultado final se divide la suma total de los resultados en cada consulta, para el numero de participaciones que cuenta cada tipo, si es el caso de en Disco 21, Mixtas 8 y en Memoria 21.

Database Oracle 12c				
Indicadores	N° Query	En Disco	Mixtas	En Memoria
Coincidencias exactas	(Q1)	1,5	-	3
	(Q2)	2	1	3
Acceso ordenado	(Q3)	1,5	-	3
	(Q4)	1,5	-	3
Casting	(Q5)	1,5	-	3
Expresiones regulares	(Q6)	1,5	-	3
	(Q7)	1	2	3
Encontrar referencias	(Q8)	1	2	3
	(Q9)	2	1	3
Select anidado	(Q10)	1	2	3
Joins	(Q11)	1	2	3
	(Q12)	2	1	3
	(Q13)	1	2	3
Búsqueda por índice	(Q14)	1,5	-	3
Agregación	(Q15)	1,5	-	3
	(Q16)	1,5	-	3
Inserción	(Q17)	1,5	-	3
Actualización	(Q18)	3	-	1,5
	(Q19)	3	-	1,5
Eliminación	(Q20)	1,5	-	3
	(Q21)	3	-	1,5
Puntaje Final		1,67	1,63	2,79

Tabla 39: Matriz de calificación Database Oracle 12c.

(Elaboración Propia).

#### 4.4.2.3 Promedios Finales Segunda ejecución.

Para esta segunda ejecución se opta una vez más por dividir los datos en tres grupos, los cuales permitirán una mejor visualización de los mismos.

Primer grupo conformado por los datos que pueden ser tomados en los tres tipos: en Disco, Mixtas y en Memoria.

Indicadores	N° Query	En Disco	Mixtas	En Memoria
Coincidencias exactas	(Q2)	00:04,766	00:05,046	00:00,608
Expresiones regulares	(Q7)	01:38,682	01:36,609	00:41,261
Encontrar referencias	(Q8)	02:35,398	02:36,801	01:08,000
	(Q9)	00:14,030	00:14,922	00:02,751
Select anidado	(Q10)	00:42,243	00:34,035	00:06,204
Joins	(Q11)	04:45,640	05:45,561	02:10,752
	(Q12)	03:55,226	03:55,761	01:06,708
	(Q13)	03:22,740	03:43,800	00:57,248

Tabla 40: Extracto 1 En Disco vs Mixta vs En Memoria - Matriz de toma de datos Database Oracle 12c sin liberar memoria.

(Elaboración Propia).

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

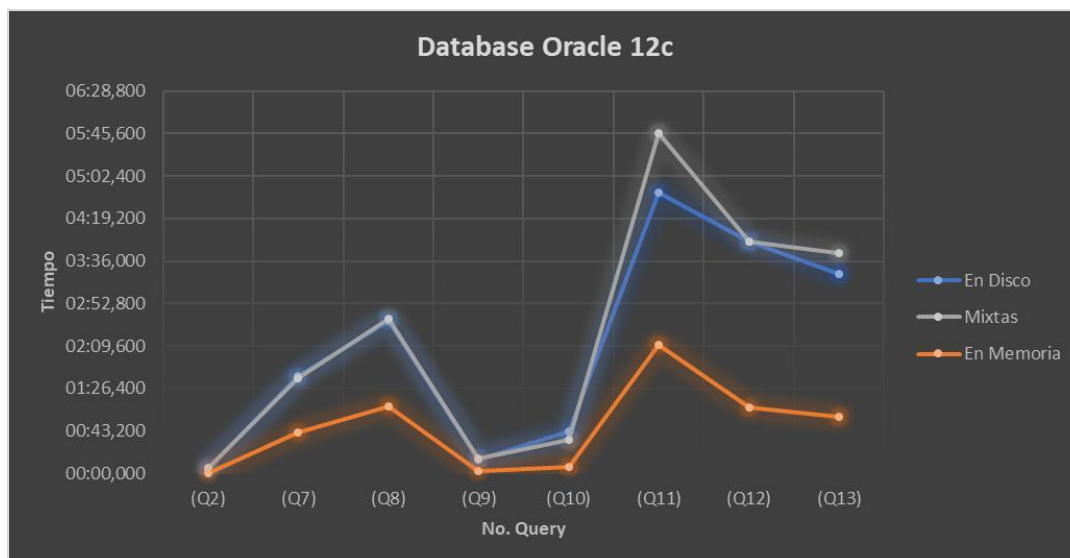


Gráfico 29: Grafico E1 - En Disco vs Mixta vs En Memoria Database Oracle 12c sin liberar memoria.

(Elaboración Propia).

Segundo grupo conformado por los datos que son viables en Disco y Memoria.

Indicadores	N° Query	En Disco	En Memoria
Acceso ordenado	(Q3)	03:27,644	01:06,312
	(Q4)	02:48,276	00:59,708
Casting	(Q5)	00:36,645	00:25,976
Expresiones regulares	(Q6)	00:49,588	00:21,412
Actualización	(Q18)	03:25,078	05:56,657
	(Q19)	03:07,529	04:05,165

Tabla 41: Extracto 2 En Disco vs En Memoria - Matriz de promedios Database Oracle 12c sin liberar memoria.

(Elaboración Propia).

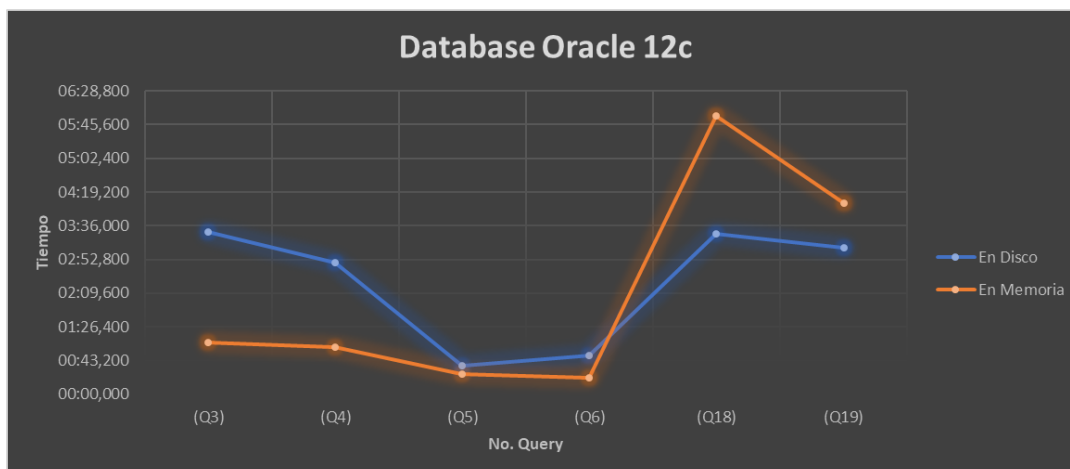


Gráfico 30: Grafico E2 - En Disco vs En Memoria Database Oracle 12c sin liberar memoria.

(Elaboración Propia).

Tercer grupo, los datos que son viables en Disco y en Memoria y que además se diferencian mucho en tiempo en cuanto a los del segundo grupo.

Indicadores	N° Query	En Disco	En Memoria
Coincidencias exactas	(Q1)	00:03,547	00:00,328
Búsqueda por índice	(Q14)	00:07,586	00:01,766
Agregación	(Q15)	00:00,735	00:00,594
	(Q16)	00:11,453	00:00,125
Inserción	(Q17)	00:01,417	00:00,971

Tabla 42: Extracto 3 En Disco vs En Memoria - Matriz de promedios Database Oracle 12c.

(Elaboración Propia).

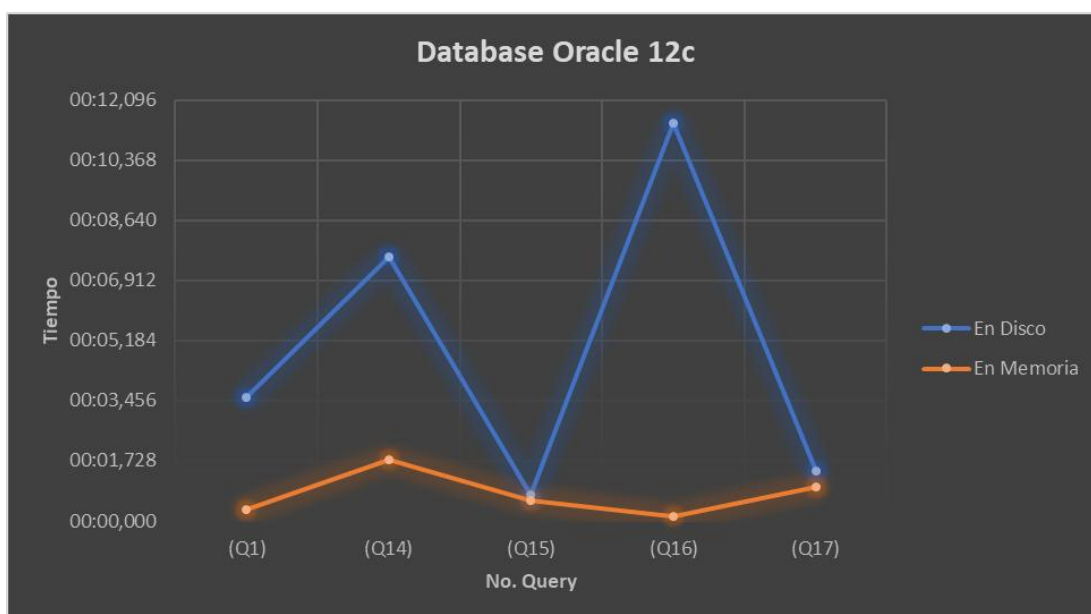


Gráfico 31: Gráfico E3 - En Disco vs En Memoria Database Oracle 12c.

(Elaboración Propia).

Nota: Las Eliminaciones no fueron tomadas en cuenta por la misma razón que en SQL Server, el hecho de que no se puede eliminar el mismo elemento en dos ocasiones seguidas.

#### 4.4.2.4 Contraste Primeras ejecuciones vs Segundas ejecuciones sin liberar memoria.

Esta comparativa nos muestra los resultados de cómo opera Oracle Database 12c luego de ejecutar por primera vez una consulta y posteriormente ejecutar la misma consulta.

Primero como en el gestor anterior, podemos observar la comparación entre las primeras ejecuciones de disco y las segundas ejecuciones en disco sin liberar memoria.

Indicadores	N° Query	2da En Disco	En Disco
Coincidencias exactas	(Q1)	00:03,547	00:04,384
	(Q2)	00:04,766	00:05,007
Acceso ordenado	(Q3)	03:27,644	03:01,923
	(Q4)	02:48,276	02:43,453
Casting	(Q5)	00:36,645	00:36,197
Expresiones regulares	(Q6)	00:49,588	00:52,491
	(Q7)	01:38,682	01:46,333
Encontrar referencias	(Q8)	02:35,398	02:37,467
	(Q9)	00:14,030	00:14,169
Select anidado	(Q10)	00:42,243	00:43,502
Joins	(Q11)	04:45,640	06:07,568
	(Q12)	03:55,226	03:58,442
	(Q13)	03:22,740	03:49,624
Búsqueda por índice	(Q14)	00:07,586	00:07,382
Agregación	(Q15)	00:00,735	00:00,741
	(Q16)	00:11,453	00:11,197
Inserción	(Q17)	00:01,417	00:01,172
Actualización	(Q18)	03:25,078	04:09,426
	(Q19)	03:07,529	03:43,698

Tabla 43: Matriz Primera ejecución en Disco vs Segunda ejecución en Disco sin liberar memoria Database Oracle 12c.

(Elaboración Propia).

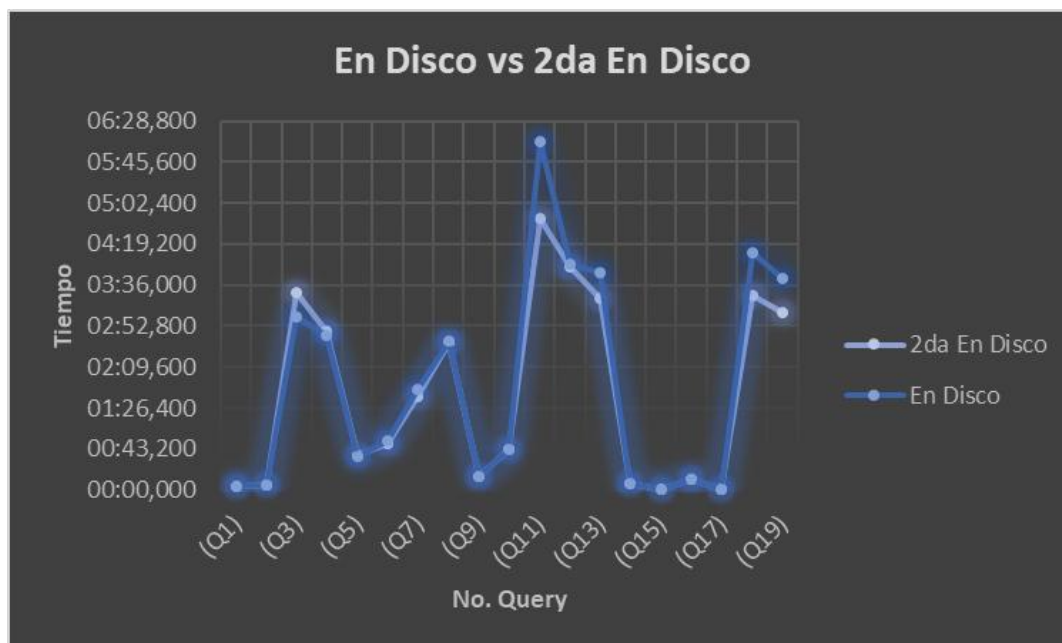


Gráfico 32: Grafico Primera ejecución en Disco vs Segunda ejecución en Disco sin liberar memoria Database Oracle 12c.

(Elaboración Propia).

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

Comparación entre las primeras ejecuciones mixtas y las segundas ejecuciones mixtas sin liberar memoria.

Indicadores	N° Query	2da Mixta	Mixta
Coincidencias exactas	(Q2)	00:05,046	00:05,216
Expresiones regulares	(Q7)	01:36,609	01:17,944
Encontrar referencias	(Q8)	02:36,801	02:31,248
	(Q9)	00:14,922	00:14,569
Select anidado	(Q10)	00:34,035	00:31,727
Joins	(Q11)	05:45,561	06:07,288
	(Q12)	03:55,761	04:00,695
	(Q13)	03:43,800	03:34,557

Tabla 44: Matriz Primera ejecución Mixta vs Segunda ejecución Mixta sin liberar memoria Database Oracle 12c.

(Elaboración Propia).

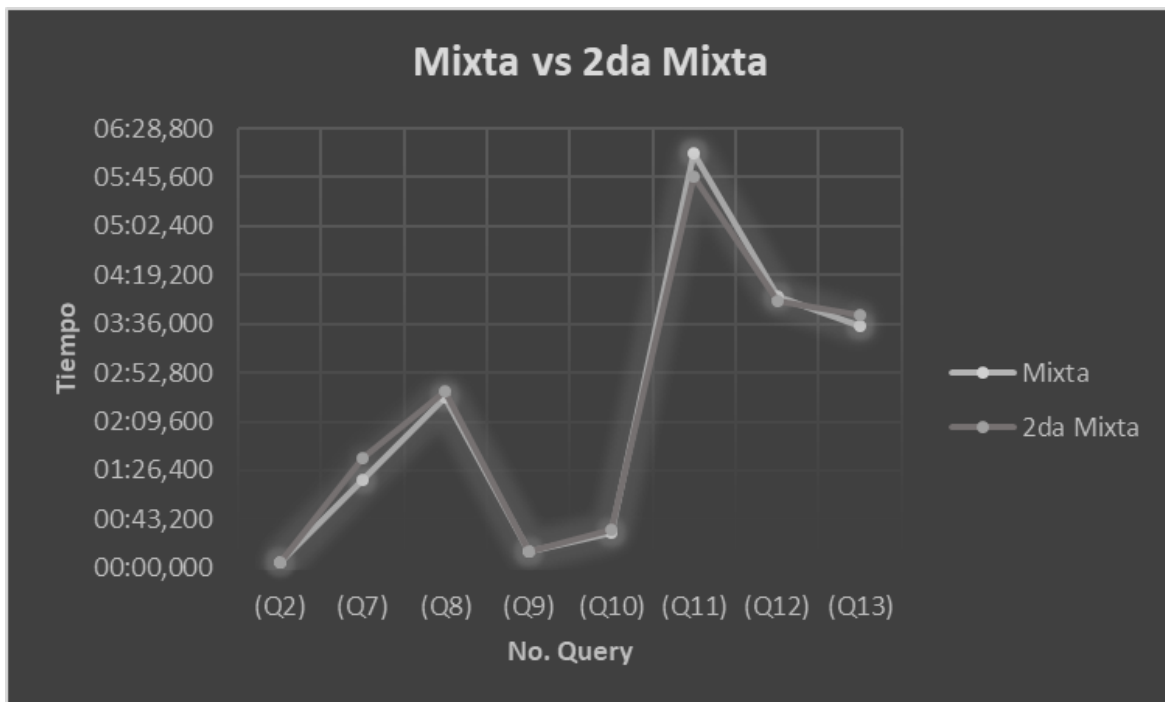


Gráfico 33: Grafico Primera ejecución Mixta vs Segunda ejecución Mixta sin liberar memoria Database Oracle 12c.

(Elaboración Propia).

Comparación entre las primeras ejecuciones en memoria y las segundas ejecuciones en memoria.

Indicadores	N° Query	En Memoria	2da En Memoria
Coincidencias exactas	(Q1)	00:00,448	00:00,328
	(Q2)	00:00,656	00:00,608
Acceso ordenado	(Q3)	01:03,071	01:06,312
	(Q4)	01:02,239	00:59,708
Casting	(Q5)	00:27,091	00:25,976
Expresiones regulares	(Q6)	00:21,548	00:21,412
	(Q7)	00:43,295	00:41,261
Encontrar referencias	(Q8)	01:16,088	01:08,000
	(Q9)	00:02,732	00:02,751
Select anidado	(Q10)	00:06,069	00:06,204
Joins	(Q11)	02:18,000	02:10,752
	(Q12)	01:08,542	01:06,708
	(Q13)	01:00,799	00:57,248
Búsqueda por índice	(Q14)	00:01,769	00:01,766
Agregación	(Q15)	00:00,535	00:00,594
	(Q16)	00:00,105	00:00,125
Inserción	(Q17)	00:01,107	00:00,971
Actualización	(Q18)	05:16,309	05:56,657
	(Q19)	04:17,374	04:05,165

Tabla 45: Matriz Primera ejecución en Memoria vs Segunda ejecución en Memoria Database Oracle 12c.

(Elaboración Propia).

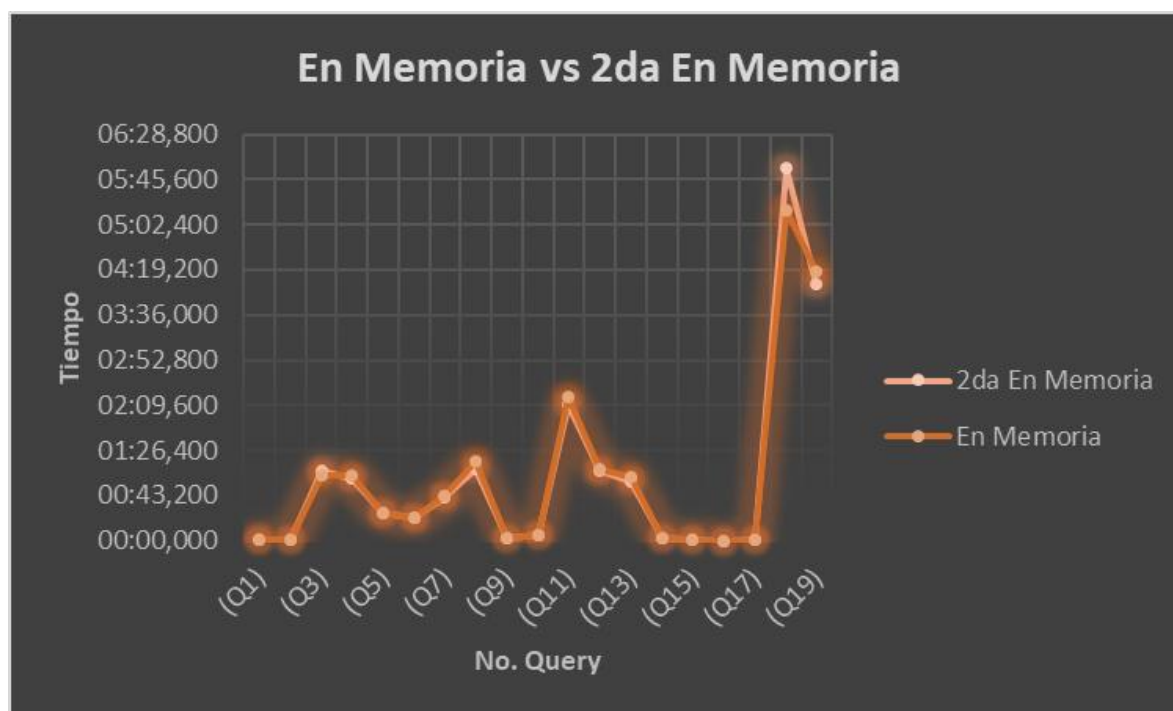


Gráfico 34: Grafico Primera ejecución en Memoria vs Segunda ejecución en Memoria Database Oracle 12c.

(Elaboración Propia).

Y por último como en el caso del gestor anterior, es llamativo comparar la segunda ejecución en disco vs la forma en memoria, para comprobar si es que como en el caso del anterior gestor, Database Oracle 12c iguala o mejora el rendimiento en la segunda ejecución en disco.

Indicadores	N° Query	2da En Disco	En Memoria
Coincidencias exactas	(Q1)	00:03,547	00:00,448
	(Q2)	00:04,766	00:00,656
Acceso ordenado	(Q3)	03:27,644	01:03,071
	(Q4)	02:48,276	01:02,239
Casting	(Q5)	00:36,645	00:27,091
Expresiones regulares	(Q6)	00:49,588	00:21,548
	(Q7)	01:38,682	00:43,295
Encontrar referencias	(Q8)	02:35,398	01:16,088
	(Q9)	00:14,030	00:02,732
Select anidado	(Q10)	00:42,243	00:06,069
Joins	(Q11)	04:45,640	02:18,000
	(Q12)	03:55,226	01:08,542
	(Q13)	03:22,740	01:00,799
Búsqueda por índice	(Q14)	00:07,586	00:01,769
Agregación	(Q15)	00:00,735	00:00,535
	(Q16)	00:11,453	00:00,105
Inserción	(Q17)	00:01,417	00:01,107
Actualización	(Q18)	03:25,078	05:16,309
	(Q19)	03:07,529	04:17,374

Tabla 46: Matriz Primera ejecución en Memoria vs Segunda ejecución en Disco SQL Database Oracle 12c.

(Elaboración Propia).



Gráfico 35: Gráfico Primera ejecución en Memoria vs Segunda ejecución en Disco Database Oracle 12c.

(Elaboración Propia).

### 4.4.3 Oracle TimesTen.

El gestor TimesTen al no tener las opciones Mixta y en Disco, no cuenta con más información para realizar más análisis o comparaciones dentro del mismo gestor, pero sus datos de promedios en memoria servirán para los posteriores contrastes con los otros gestores de bases de datos.

#### 4.4.3.1 Promedios Finales.

Times Ten		
Indicadores	N° Query	En Memoria
Coincidencias exactas	(Q1)	00:01,631
	(Q2)	00:03,787
Acceso ordenado	(Q3)	02:42,038
	(Q4)	02:41,603
Casting	(Q5)	01:29,911
Expresiones regulares	(Q6)	01:16,025
	(Q7)	01:59,267
Encontrar referencias	(Q8)	02:01,621
	(Q9)	00:02,022
Select anidado	(Q10)	00:26,532
Joins	(Q11)	00:50,006
	(Q12)	03:12,127
	(Q13)	02:41,115
Búsqueda por índice	(Q14)	00:06,437
Agregación	(Q15)	00:01,650
	(Q16)	00:00,388
Inserción	(Q17)	00:02,438
Actualización	(Q18)	00:00,825
	(Q19)	00:09,106
Eliminación	(Q20)	00:49,944
	(Q21)	00:38,306

Tabla 47: Matriz de promedios finales TimesTen.

(Elaboración Propia).

### 4.4.4 Contraste Bases de Datos.

#### 4.4.4.1 Bases de Datos en Disco

Comenzamos los contrastes entre bases de datos con la forma en disco. Para esto tomamos los promedios finales en disco de los gestores SQL Server 2016 y Database Oracle 12c, que si tienen la opción de manejo de bases de datos en Disco.

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

Bases de Datos en Disco			
Indicadores	N° Query	SQL Server 2016	Database Oracle 12c
Coincidencias exactas	(Q1)	00:30,378	00:04,384
	(Q2)	00:38,613	00:05,007
Acceso ordenado	(Q3)	02:14,356	03:01,923
	(Q4)	01:53,983	02:43,453
Casting	(Q5)	01:48,441	00:36,197
Expresiones regulares	(Q6)	00:14,659	00:52,491
	(Q7)	00:54,262	01:46,333
Encontrar referencias	(Q8)	01:00,153	02:37,467
	(Q9)	01:36,093	00:14,169
Select anidado	(Q10)	02:10,372	00:43,502
Joins	(Q11)	01:17,105	06:07,568
	(Q12)	00:57,141	03:58,442
	(Q13)	01:55,074	03:49,624
Búsqueda por índice	(Q14)	00:27,650	00:07,382
Agregación	(Q15)	00:09,662	00:00,741
	(Q16)	01:25,357	00:11,197
Inserción	(Q17)	00:00,333	00:01,172
Actualización	(Q18)	09:01,531	04:09,426
	(Q19)	26:18,021	03:43,698
Eliminación	(Q20)	01:35,737	03:52,980
	(Q21)	01:40,977	01:56,513

Tabla 48: Matriz de Contraste de Bases de Datos en Disco - Database Oracle 12c vs SQL Server 2016.

(Elaboración Propia).

Las calificaciones fueron ponderadas dando el total de los puntos o la mitad respectivamente, ya que solo dos gestores intervienen en este contraste.

Bases de Datos en Disco			
N° Query	Indicadores	SQL Server 2016	Database Oracle 12c
(Q1)	Consultas	1,5	3
(Q2)		1,5	3
(Q3)		3	1,5
(Q4)		3	1,5
(Q5)		1,5	3
(Q6)		3	1,5
(Q7)		3	1,5
(Q8)		3	1,5
(Q9)		1,5	3
(Q10)		1,5	3
(Q11)		3	1,5
(Q12)		3	1,5
(Q13)		3	1,5
(Q14)		1,5	3
(Q15)		1,5	3
(Q16)		1,5	3
<b>SubTotal</b>		36	36
(Q17)	Inserción	3	1,5
<b>SubTotal</b>		3	1,5
(Q18)	Actualización	1,5	3
(Q19)		1,5	3
<b>SubTotal</b>		3	6
(Q20)	Eliminación	3	1,5
(Q21)		3	1,5
<b>SubTotal</b>		6	3
<b>Resultado</b>		48	46,5

Tabla 49: Matriz de Calificación de Contraste de Bases de Datos en Disco - Database Oracle 12c vs SQL Server 2016.

(Elaboración Propia).

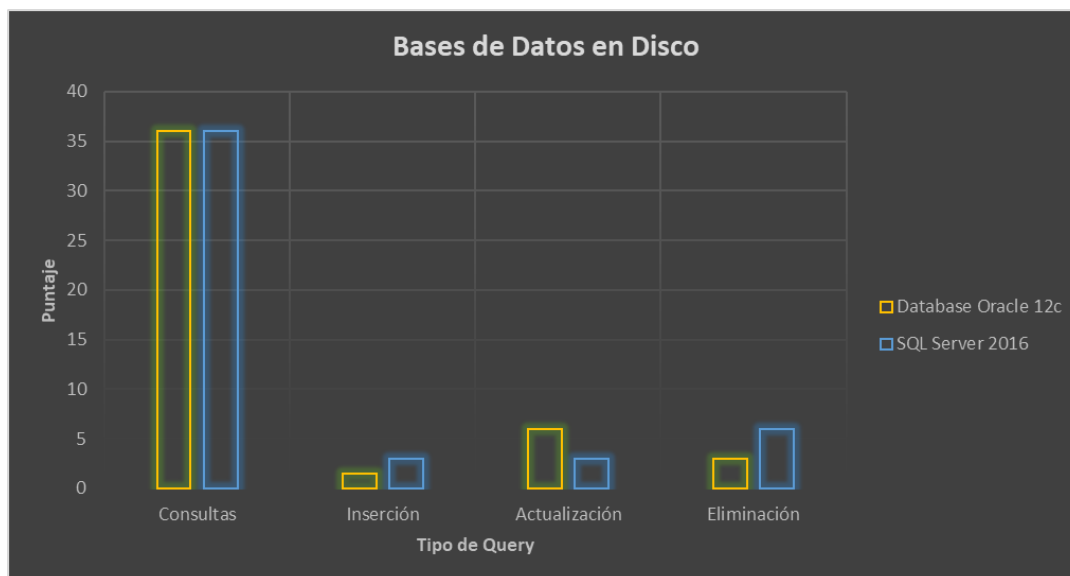


Gráfico 36: Gráfico de Contraste de Bases de Datos en Disco - Database Oracle 12c vs SQL Server 2016.

(Elaboración Propia).

#### 4.4.4.2 Bases de Datos Mixtas

Para el contraste de bases de datos mixtas tomamos los promedios finales de los gestores en forma mixta, además se toma solo las consultas que se pueden realizar de forma mixta en SQL Server 2016 y Database Oracle 12c que son los gestores que permiten el manejo de bases de datos mixtas.

Bases de Datos Mixtas			
Indicadores	N° Query	SQL Server 2016	Database Oracle 12c
Coincidencias exactas	(Q2)	00:30,583	00:05,216
Expresiones regulares	(Q7)	00:42,350	01:17,944
Encontrar referencias	(Q8)	00:50,976	02:31,248
	(Q9)	01:32,023	00:14,569
Select anidado	(Q10)	02:26,270	00:31,727
Joins	(Q11)	01:02,476	06:07,288
	(Q12)	00:46,279	04:00,695
	(Q13)	01:49,788	03:34,557

Tabla 50: Matriz de Contraste de Bases de Datos Mixtas - Database Oracle 12c vs SQL Server 2016.

(Elaboración Propia).

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

Las calificaciones se ponderaron tomando los mismos criterios que se siguieron con anterioridad en las matrices de calificaciones anteriormente realizadas.

Bases de Datos Mixtas			
N° Query	Indicadores	SQL Server 2016	Database Oracle 12c
(Q2)	Consultas	1,5	3
(Q7)		3	1,5
(Q8)		3	1,5
(Q9)		1,5	3
(Q10)		1,5	3
(Q11)		3	1,5
(Q12)		3	1,5
(Q13)		3	1,5
<b>Resultado</b>		<b>19,5</b>	<b>16,5</b>

Tabla 51: Matriz de Calificación de Contraste de Bases de Datos Mixtas - Database Oracle 12c vs SQL Server 2016.

(Elaboración Propia).

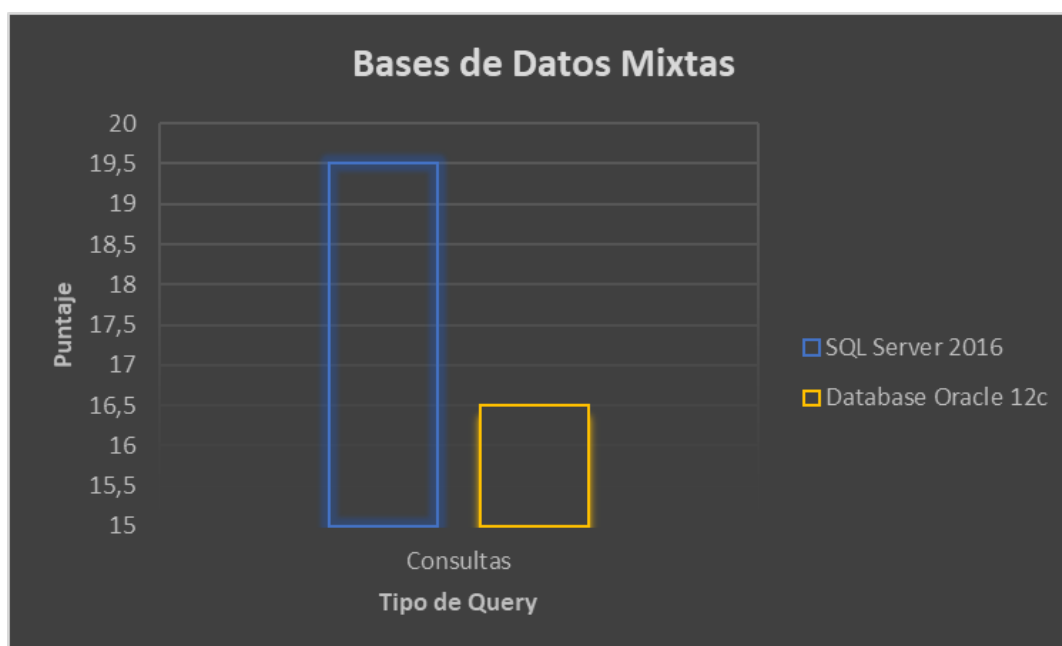


Gráfico 37: Gráfico de Contraste de Bases de Datos Mixtas - Database Oracle 12c vs SQL Server 2016.

(Elaboración Propia).

#### 4.4.4.3 Bases de Datos en Memoria

El contraste de bases de datos en memoria es el único que toma información de los tres gestores de bases de datos y por esto se puede visualizar mejor los resultados obtenidos en cada tipo de consulta respecto a las demás.

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

Bases de Datos en Memoria				
Indicadores	N° Query	Times Ten	SQL Server 2016	Database Oracle 12c
Coincidencias exactas	(Q1)	00:01,631	00:02,418	00:00,448
	(Q2)	00:03,787	00:10,182	00:00,656
Acceso ordenado	(Q3)	02:42,038	00:23,959	01:03,071
	(Q4)	02:41,603	00:22,002	01:02,239
Casting	(Q5)	01:29,911	00:11,321	00:27,091
Expresiones regulares	(Q6)	01:16,025	00:07,852	00:21,548
	(Q7)	01:59,267	00:14,031	00:43,295
Encontrar referencias	(Q8)	02:01,621	00:19,260	01:16,088
	(Q9)	00:02,022	00:01,227	00:02,732
Select anidado	(Q10)	00:26,532	00:14,431	00:06,069
Joins	(Q11)	00:50,006	00:35,982	02:18,000
	(Q12)	03:12,127	00:21,401	01:08,542
	(Q13)	02:41,115	00:17,443	01:00,799
Búsqueda por índice	(Q14)	00:06,437	00:02,857	00:01,769
Agregación	(Q15)	00:01,650	00:01,033	00:00,535
	(Q16)	00:00,388	00:00,532	00:00,105
Inserción	(Q17)	00:02,438	00:00,035	00:01,107
Actualización	(Q18)	00:00,825	00:01,773	05:16,309
	(Q19)	00:09,106	00:19,927	04:17,374
Eliminación	(Q20)	00:49,944	00:01,631	01:23,301
	(Q21)	00:38,306	00:01,623	02:20,538

Tabla 52: Matriz de Contraste de Bases de Datos en Memoria - Database Oracle 12c vs SQL Server 2016 vs TimesTen.

(Elaboración Propia).

Bases de Datos en Memoria				
N° Query	Indicadores	Times Ten	SQL Server 2016	Database Oracle 12c
(Q1)	Consultas	2	1	3
(Q2)		2	1	3
(Q3)		1	3	2
(Q4)		1	3	2
(Q5)		1	3	2
(Q6)		1	3	2
(Q7)		1	3	2
(Q8)		1	3	2
(Q9)		2	3	1
(Q10)		1	2	3
(Q11)		2	3	1
(Q12)		1	3	2
(Q13)		1	3	2
(Q14)		1	2	3
(Q15)		1	2	3
(Q16)		2	1	3
<b>SubTotal</b>		21	39	36
(Q17)	Inserción	1	3	2
<b>SubTotal</b>		1	3	2
(Q18)	Actualización	3	2	1
(Q19)		3	2	1
<b>SubTotal</b>		6	4	2
(Q20)	Eliminación	2	3	1
(Q21)		2	3	1
<b>SubTotal</b>		4	6	2
<b>Resultado</b>		32	52	42

Tabla 53: Matriz de Calificación de Contraste de Bases de Datos en Memoria -Database Oracle 12c vs SQL Server 2016 vs TimesTen.

(Elaboración Propia).

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

En este Grafico podemos visualizar los resultados de una forma un poco más global que indican el desempeño de los tres gestores en Disco, Mixta y en Memoria.

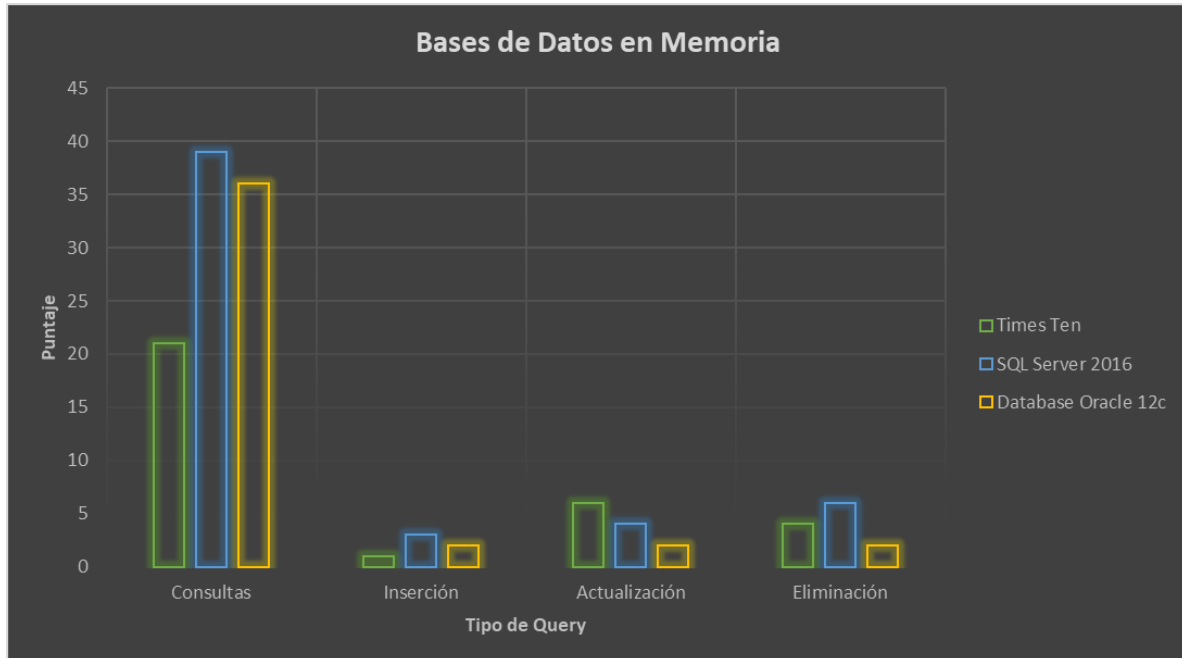


Gráfico 38: Grafico Contraste de Bases de Datos en Memoria- Database Oracle 12c vs SQL Server 2016 vs TimesTen.

(Elaboración Propia).

## 5 Capítulo: Conclusiones, Recomendaciones

### 5.1 Conclusiones del análisis benchmark.

#### 5.1.1 Conclusiones SQL Server 2016.

- Cuando tenemos en cuenta las consultas que se realizaron en los 3 gestores y sus 3 formas de manejo de datos (Disco, Mixta y Memoria), podemos observar que en este caso el tipo en memoria obtiene tiempos de respuesta mucho menores, superando ampliamente a disco y mixtas.
- Con respecto a los resultados del tipo mixta, los mismos se asemejan mucho a los resultados en disco, disminuyendo muy poco los tiempos de las consultas, incluso en la consulta 10 que corresponde a los select anidados obtiene un valor más elevado.
- Cuando analizamos los resultados de las consultas que solo se evalúan en los tipos (Disco y Memoria), correspondientes al segundo grupo de datos, los resultados son muy claros, memoria obtiene menores tiempos, haciendo énfasis en los valores referentes a Acceso ordenado y casting.
- También se observa que, en el caso de las actualizaciones y eliminaciones correspondientes al tercer grupo de datos, existe una diferencia marcada, ya que se obtuvieron tiempos muy altos en la forma en disco, y en la forma de memoria tiempos muy inferiores por ende mejores.
- Con respecto a las segundas ejecuciones en las tres formas de manejo de datos (Disco, Mixta y Memoria), se obtienen valores bastante llamativos si se compara con los valores referentes a la forma en disco, ya que el gestor no libera la memoria y obtiene un mejor rendimiento en la mayoría de las consultas.
- Además, en las segundas ejecuciones obtenemos que, en la mayoría de las consultas estas no se diferencian en mucho entre sí, segunda ejecución en disco vs segunda ejecución mixta, vs segunda ejecución en memoria, los resultados son estables salvo el caso de los select anidados o consulta 10 y consulta 2 correspondiente a coincidencias exactas.

- Como resultado podemos concluir que SQL server en su forma en memoria, obtiene los tiempos de respuesta más, casi comparables con las segundas ejecuciones en disco (si se realiza la misma consulta consecutiva), la siguiente su forma mixta y con muy poca diferencia su forma en disco.
- No se debe dejar de lado los resultados que se obtuvieron en la segunda ejecución, ya que esto puede darle un valor añadido para las consultas que son consecutivas dentro de una base de datos en disco, sin olvidar que solo funciona si la consulta es la misma y consecutiva.
- En cuanto a su manejo y experiencia como usuario del gestor, SQL Server brinda muchas facilidades con T-SQL, para sus consultas, también se debe notar que los resultados y el mismo gestor es muy estable.

### **5.1.2 Conclusiones Database Oracle 12c.**

- En el caso de este gestor podemos observar que en el primer grupo de datos correspondientes a la comparativa entre (Disco, Mixta y Memoria), tenemos un comportamiento parecido que con el anterior gestor. Los tiempos de memoria son menores y por conclusión mejores.
- Cuando vemos el contraste de tiempos en disco y mixta podemos observar que no se tienen mayor diferencia entre sí.
- Con respecto al segundo grupo de datos correspondientes a la comparativa de disco y memoria, observamos que en memoria se obtienen mejores resultados, especialmente en las consultas 3 y 4 correspondientes al acceso ordenado, y una mayor diferencia de tiempo en la consulta 16 del indicador agregación.
- Observando los resultados del tercer grupo de datos correspondientes a la comparativa de disco y memoria, se observa una diferencia curiosa en las consultas de actualización, que muestra un mejor desempeño en disco que en memoria, concluyendo que se realizan en menor tiempo las actualizaciones estando en disco contra las que están en memoria.
- Analizando las segundas ejecuciones se puede ver que, a diferencia del anterior gestor, en Database Oracle 12c si se libera la memoria, ya que los resultados obtenidos en las segundas ejecuciones son similares a los

iniciales en disco y por esto su rendimiento en la mayoría de las consultas se mantiene muy parecido.

- Al final observamos que el manejo en memoria es el que mejores tiempos obtiene tanto en las primeras ejecuciones como en las segundas ejecuciones, pero su dominio no es total y tan marcado como el caso del anterior gestor.
- En cuanto a su manejo y experiencia como usuario del gestor, al poder accederlo desde Oracle SQL Developer se vuelve más intuitivo y sencillo de manejar. Además, el uso de PL/SQL, hace que muchas consultas se puedan realizar más fácilmente o incluso llegar a optimizarlas.

### **5.1.3 Conclusiones TimesTen.**

- En el caso de TimesTen al ser un gestor que opera solo en memoria, las comparativas en cuanto a los resultados se vuelven un poco complicados.
- Este gestor tiene un rendimiento bueno con las consultas de actualización, obteniendo tiempos bajos.
- Además, en TimesTen, se puede notar que las eliminaciones le cuestan mucho y produce un rendimiento muy bajo, ya que se obtuvieron tiempos de respuesta muy altos.
- En cuanto a su manejo y experiencia como usuario del gestor, de la misma forma que en Oracle 12c, TimesTen se puede manejar desde Oracle SQL Developer, ayudando al usuario de igual manera que en el gestor anterior. Además, tiene algunas variaciones al lenguaje Oracle PL/SQL, pero con sus alternativas de sintaxis para TimesTen, se lograron cubrir todas las consultas del análisis.

### **5.1.4 Conclusiones Contraste Bases de Datos en Memoria**

- Esta comparativa evidencia, como dato importante a destacar, que visto de una manera global, el gestor que maneja sus datos solo en memoria de forma nativa (TimesTen), es la que mayores tiempos de respuesta arroja, y por esto se obtuvo un menor rendimiento.

- TimesTen Obtiene los mejores resultados solo en las consultas 18 y 19 correspondientes al indicador de actualizaciones, las cuales claramente las maneja de una manera optimizada.
- La base que mejores o menores tiempos arroja en memoria es SQL Server 2016, obteniendo resultados ampliamente mejores que TimesTen y también mejores que Database Oracle 12c.
- SQL Server mantiene una estabilidad en cuanto a sus tiempos, cosa que no pasaba con los otros gestores que obtenían resultados muy lejanos o variados en las diferentes consultas.
- En cuanto al contraste de gestores de base de datos mixtas podemos observar que SQL Server 2016 supera en 5 de las 8 consultas a Database Oracle 12c, y la superioridad más notable se encuentra en las consultas del tipo join que SQL Server las realiza en mucho menos tiempo.
- Por último, se puede visualizar con los gestores de bases de datos en disco, que resultados muy parecidos, a pesar de que SQL Server consigue mejores resultados, la superioridad no es notable y se mantiene muy parejo con Database Oracle 12c en disco.

## **5.2 Conclusiones de la Disertación.**

- Podemos concluir que, como demuestra esta comparativa el manejo de base de datos alojadas en la memoria principal, si mejora el rendimiento de la mayoría de las consultas, pero también tiene mucho que ver el gestor con el que se está trabajando, para obtener un rendimiento considerablemente mejor.
- El actualizar una base de datos en disco para que se encuentre en forma mixta o en memoria es sencillo, pero se debe analizar el espacio en memoria a ocupar y el momento adecuado para realizarlo.
- Es lo mejor manejar las bases de datos en memoria, ya que actualmente los costos de la memoria principal se han reducido, volviendo factible su implementación.

- Si las consultas que se realizan dentro de una base de datos en disco son muy repetitivas, se puede obtener un muy buen rendimiento con las segundas ejecuciones sin liberar memoria en SQL Server 2016.
- El mejor Gestor de Bases de datos según esta comparativa, por estabilidad, facilidad de configuración y mejor rendimiento es SQL Server 2016.
- El peor Gestor según esta comparativa curiosamente es Oracle TimesTen que a pesar de ser principalmente creada para operar en memoria no fue superior que los otros gestores en modo memoria.
- Como se pudo demostrar en esta disertación de grado, el probar la utilidad de nuevas tecnologías o tecnologías no muy utilizadas, puede llegar a darnos muchas y nuevas herramientas que mejoren en varios aspectos nuestras tareas.
- El ambiente de pruebas fue en un sistema operativo Windows, este es el ambiente óptimo para SQL Server y esto se considera determinante para los resultados obtenidos que claramente fueron mejores con respecto a los otros gestores de Oracle.
- Se debe tomar en cuenta que por su forma de procesamiento y ejecución de las consultas cada uno de los gestores de bases de datos arrojó diferentes tiempos entre sí, por lo tanto, se debe realizar un análisis en cuanto al tipo de consultas que se va a realizar con mayor frecuencia sobre la base de datos, ya que este puede ser un factor para tomar en cuenta si se busca escoger un motor de base de datos.

### **5.3 Recomendaciones.**

- Se debe tener muy en cuenta el tamaño de la base de datos a alojar en la memoria principal, para que, de esta forma no se tengan inconvenientes referentes al espacio disponible en memoria principal.
- La dificultad que existe al implementar una base de datos en memoria ha disminuido mucho, y por esto si se cuenta con el espacio necesario en memoria principal se debe realizar esta implementación.
- Si se maneja Oracle se debe tener un especial cuidado ya que, si no se deja el espacio suficiente o no se conoce la forma de asignación de memoria de este gestor para cada uno de sus componentes para el

funcionamiento del mismo, este se bloquea y queda inoperativo, complicando mucho su reparación.

- Antes de implementar cualquiera de estas tres formas de manejo de datos, se debe realizar un análisis completo, de la base de datos con la cual se va a operar, porque en algunos casos el rendimiento puede ser muy optimizado y en otros podría llegar a ser innecesario.
- Si no se desea o no se puede implementar bases de datos en memoria, se puede optar por su forma mixta que requiere menos espacio en la memoria principal y puede ser enfocada a la información más utilizada o crítica dentro de la base de datos.
- Para futuros trabajos se podría tener en cuenta más indicadores como el consumo de los recursos del computador, que pueden complementar la comparativa, dotando al análisis benchmark de mucha más información.
- Además, se recomienda que, se puede complementar el análisis de esta disertación, realizando las mismas pruebas en un ambiente propio de Oracle, para lograr un mayor contraste de esta información.

## 6 Anexos y Bibliografía

### 6.1 Definición de Queries

- **Coincidencias exactas**

(Q1) Muestra atributos de cliente de registros que contengan en su nombre PA.

(Q2) Muestra atributos de cliente y venta de ventas realizadas en el 2004.

- **Acceso ordenado**

(Q3) Lista de carros del más barato al más costoso.

(Q4) Lista de carros del más costoso al más barato.

- **Casting**

(Q5) Muestra el precio convertido en un dato tipo char como mensaje sin decimales.

- **Expresiones regulares**

(Q6) Muestra las fechas realizadas de todos los años de los meses enero a junio.

(Q7) Muestra información de dos tablas cliente y venta de ventas de antes del año 2000.

- **Encontrar referencias**

(Q8) Lista de los nombres de las personas y el número de carros que compro.

(Q9) Lista de las marcas de carros comprados y la cantidad comprada.

- **Select Anidado**

(Q10) Presenta los datos del cliente junto con los datos del vehículo que ha comprado, pero de aquellos menores a \$50000.

- **Joins**

(Q11) Permite combinar registros por medio de full outer join.

(Q12) Permite combinar registros por medio de inner join.

(Q13) Permite combinar registros por medio de right join.

- **Búsqueda por índice**

(Q14) Se crea un índice sobre el atributo edad de la tabla cliente y posteriormente se realiza la consulta de los clientes mayores a 40 años

- **Agregación**

(Q15) Muestra el promedio de las ventas agrupadas por fechas.

(Q16) Se visualiza la suma de los costos de los carros y sus cantidades por marca.

- **Inserción**

(Q17) Realiza inserciones múltiples a la base de datos.

- **Actualización**

(Q18) Se aumenta un el costo de los carros de marca Audi multiplicando por 1.10.

(Q19) Se aumenta un el costo de todos los carros multiplicando por 1.05.

- **Eliminación**

(Q20) Se elimina todos los registros de ventas realizadas antes del 2000.

(Q20) Se elimina todos los registros de la tabla Venta.

## 6.2 Queries SQL Server 2016

### (Q1)

```
Select cli_Nombre, cli_Apellido, cli_Telefono  
From Cliente  
Where cli_Nombre like '%Pa%';
```

### (Q2)

```
Select cli_Nombre, ven_Fecha, ven_CostoTotal  
from Cliente, Venta  
where fk_Cliente = cli_Cedula and  
ven_Fecha like '%2004%';
```

### (Q3)

```
Select car_Codigo,car_Marca,car_Modelo,car_Costo  
FROM Carro  
ORDER BY car_Costo ASC
```

### (Q4)

```
Select car_Codigo,car_Marca,car_Modelo,car_Costo  
FROM Carro  
ORDER BY car_Costo DESC
```

### (Q5)

```
Select car_Modelo,'El precio es ' + CAST(CAST(car_Costo AS int) AS char(10))  
AS PrecioRango  
FROM Carro  
WHERE car_Costo BETWEEN 15000 AND 200000;
```

### (Q6)

```
Select ven_Codigo, ven_CostoTotal, fk_Cliente, ven_Fecha  
from Venta  
where ven_Fecha like '%-[0][0-6]-%';
```

**(Q7)**

```
Select cli_Nombre, ven_Fecha, ven_CostoTotal  
from Cliente, Venta  
where fk_Cliente = cli_Cedula and  
ven_Fecha like '[0-1]%';
```

**(Q8)**

```
Select cli_Nombre, cli_Apellido, COUNT(fk_Cliente) as Cantidad  
FROM Cliente, Venta  
WHERE cli_Cedula = fk_Cliente  
Group BY fk_Cliente, cli_Nombre, cli_Apellido
```

**(Q9)**

```
Select car_Marca, COUNT(car_Marca) as Cantidad  
FROM Carro, Venta  
WHERE car_Codigo = fk_Carro  
Group BY car_Marca
```

**(Q10)**

```
Select      Cliente.cli_Nombre,      cli_Apellido,      Vehiculo.car_Marca,  
Vehiculo.car_Modelo, Vehiculo.car_Costo  
from (select car_Marca, car_Modelo, car_Costo, fk_Cliente  
      from Carro, Venta  
      where car_Codigo = fk_Carro) as Vehiculo, Cliente  
where cli_Cedula = Vehiculo.fk_Cliente and Vehiculo.car_Costo <= 50000
```

**(Q11)**

```
Select cli_Nombre, cli_Apellido, cli_Cedula, cli_Telefono, ven_Fecha,  
ven_CostoTotal  
FROM Cliente FULL OUTER JOIN Venta  
ON Cliente.cli_Cedula = Venta.fk_Cliente;
```

**(Q12)**

```
Select cli_Nombre, cli_Apellido, cli_Cedula, cli_Telefono, ven_Fecha,  
ven_CostoTotal  
FROM Cliente INNER JOIN Venta
```

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

ON Cliente.cli\_Cedula = Venta.fk\_Cliente;

**(Q13)**

Select car\_Marca, car\_Modelo, ven\_Fecha, ven\_CostoTotal

FROM Carro RIGHT JOIN Venta

ON Carro.car\_Codigo = Venta.fk\_Carro;

**(Q14)**

- En Disco

Create index INDX\_CliEdad

ON Cliente (cli\_Apellido);

Select cli\_Nombre, cli\_Apellido

From Cliente

where cli\_Apellido like '%z%';

- En memoria.

ALTER TABLE Cliente ADD INDEX INDX\_CliApe (cli\_Apellido)

Select cli\_Nombre, cli\_Apellido

From Cliente

where cli\_Apellido like '%z%';

**(Q15)**

Select ven\_Fecha, AVG(ven\_CostoTotal) as PromCT

FROM Venta

GROUP BY ven\_Fecha;

**(Q16)**

Select car\_Marca, SUM(car\_Costo) as SumaTotal, COUNT(car\_Marca) as  
Cantidad

FROM Carro

GROUP BY car\_Marca;

**(Q17)**

INSERT INTO Cliente([cli\_Cedula], [cli\_Nombre], [cli\_Apellido], [cli\_Telefono],  
[cli\_Edad]) VALUES('1631121046499','Sonya','Caldwell','(574) 947-3086',40);

**(Q18)**

UPDATE Carro

SET car\_Costo = car\_Costo\*1.12

```
WHERE car_Marca = 'Audi';
```

**(Q19)**

```
UPDATE Carro
```

```
SET car_Costo = car_Costo*1.05;
```

**(Q20)**

```
DELETE FROM Venta
```

```
WHERE ven_Fecha <= '2000-01-01 00:01:10';
```

**(Q21)**

```
DELETE FROM Venta
```

```
// no es necesario WHERE 1=1
```

### 6.3 Queries Oracle 12c

**(Q1)**

```
Select cli_Nombre, cli_Apellido, cli_Telefono
```

```
from Cliente
```

```
where cli_Nombre like '%Pa%';
```

**(Q2)**

```
Select cli_Nombre, ven_Fecha, ven_CostoTotal
```

```
from Cliente, Venta
```

```
where fk_Cliente = cli_Cedula and
```

```
ven_Fecha like '%2004%';
```

**(Q3)**

```
Select car_Codigo,car_Marca,car_Modelo,car_Costo
```

```
FROM Carro
```

```
ORDER BY car_Costo ASC;
```

**(Q4)**

```
Select car_Codigo,car_Marca,car_Modelo,car_Costo
```

```
FROM Carro
```

```
ORDER BY car_Costo DESC;
```

**(Q5)**

```
Select car_Modelo,'El precio es ' || CAST(CAST(car_Costo AS number(10)) AS  
char(10)) AS PrecioRango
```

```
FROM Carro
```

```
WHERE car_Costo BETWEEN 15000 AND 200000;
```

**(Q6)**

```
Select ven_Codigo, ven_CostoTotal, fk_Cliente, ven_Fecha
from Venta
where REGEXP_LIKE (ven_fecha, '-0[0-6]-');
```

**(Q7)**

```
Select cli_Nombre, ven_Fecha, ven_CostoTotal
from Cliente, Venta
where fk_Cliente = cli_Cedula and REGEXP_LIKE (ven_fecha, '^1');
```

**(Q8)**

```
Select cli_Nombre, cli_Apellido, COUNT(fk_Cliente) as Cantidad
FROM Cliente, Venta
WHERE cli_Cedula = fk_Cliente
Group BY fk_Cliente, cli_Nombre, cli_Apellido;
```

**(Q9)**

```
Select car_Marca, COUNT(car_Marca) as Cantidad
FROM Carro, Venta
WHERE car_Codigo = fk_Carro
Group BY car_Marca;
```

**(Q10)**

```
Select      Cliente.cli_Nombre,      cli_Apellido,Vehiculo.car_Marca,
Vehiculo.car_Modelo, Vehiculo.car_Costo
from (select car_Marca,car_Modelo,car_Costo, fk_Cliente
      from Carro,Venta
      where car_Codigo = fk_Carro) Vehiculo, Cliente
where cli_Cedula = Vehiculo.fk_Cliente and Vehiculo.car_Costo <= 50000;
```

**(Q11)**

```
Select  cli_Nombre,  cli_Apellido,  cli_Cedula,  cli_Telefono,  ven_Fecha,
ven_CostoTotal
FROM Cliente FULL OUTER JOIN Venta
ON Cliente.cli_Cedula = Venta.fk_Cliente;
```

**(Q12)**

```
Select  cli_Nombre,  cli_Apellido,  cli_Cedula,  cli_Telefono,  ven_Fecha,
ven_CostoTotal
FROM Cliente INNER JOIN Venta
```

ON Cliente.cli\_Cedula = Venta.fk\_Cliente;

**(Q13)**

Select car\_Marca, car\_Modelo, ven\_Fecha, ven\_CostoTotal

FROM Carro RIGHT JOIN Venta

ON Carro.car\_Codigo = Venta.fk\_Carro;

**(Q14)**

Create index INDX\_CliEdad

ON Cliente (cli\_Apellido);

Select cli\_Nombre, cli\_Apellido

From Cliente

where cli\_Apellido like '%z%' or cli\_Apellido like '%Z%';

**(Q15)**

Select ven\_Fecha, AVG(ven\_CostoTotal) as PromCT

FROM Venta

GROUP BY ven\_Fecha;

**(Q16)**

Select car\_Marca, SUM(car\_Costo) as SumaTotal, COUNT(car\_Marca) as  
Cantidad

FROM Carro

GROUP BY car\_Marca;

**(Q17)**

INSERT INTO CLIENTE VALUES ('1684062861799', 'Tobias', 'Griffith',  
'1499490', 44);

**(Q18)**

UPDATE Carro

SET car\_Costo = car\_Costo\*1.12

WHERE car\_Marca = 'Audi';

**(Q19)**

UPDATE Carro

SET car\_Costo = car\_Costo\*1.05;

**(Q20)**

DELETE FROM Venta

WHERE ven\_Fecha <= '2000-01-01';

**(Q21)**

DELETE FROM Venta;

**6.4 Queries Times Ten**

**(Q1)**

```
Select cli_Nombre, cli_Apellido, cli_Telefono
from Cliente
where cli_Nombre like '%Pa%';
```

**(Q2)**

```
Select cli_Nombre, ven_Fecha, ven_CostoTotal
from Cliente, Venta
where fk_Cliente = cli_Cedula and
ven_Fecha like '%2004%';
```

**(Q3)**

```
Select car_Codigo,car_Marca,car_Modelo,car_Costo
FROM Carro
ORDER BY car_Costo ASC;
```

**(Q4)**

```
Select car_Codigo,car_Marca,car_Modelo,car_Costo
FROM Carro
ORDER BY car_Costo DESC;
```

**(Q5)**

```
Select car_Modelo,'El precio es ' || CAST(CAST(car_Costo AS number(10)) AS
char(10)) AS PrecioRango
FROM Carro
WHERE car_Costo BETWEEN 15000 AND 200000;
```

**(Q6)**

```
Select ven_Codigo, ven_CostoTotal, fk_Cliente, ven_Fecha
from Venta
where ven_fecha LIKE '%-01-%' or ven_fecha LIKE '%-02-%' or ven_fecha
LIKE '%-03-%'
or ven_fecha LIKE '%-04-%' or ven_fecha LIKE '%-05-%' or ven_fecha LIKE '%-
06-%';
```

**(Q7)**

```
Select cli_Nombre, ven_Fecha, ven_CostoTotal
from Cliente, Venta
where fk_Cliente = cli_Cedula and
ven_Fecha like '1%';
```

**(Q8)**

```
Select cli_Nombre, cli_Apellido, COUNT(fk_Cliente) as Cantidad
FROM Cliente, Venta
WHERE cli_Cedula = fk_Cliente
Group BY fk_Cliente, cli_Nombre, cli_Apellido;
```

**(Q9)**

```
Select car_Marca, COUNT(car_Marca) as Cantidad
FROM Carro, Venta
WHERE car_Codigo = fk_Carro
Group BY car_Marca;
```

**(Q10)**

```
Select      Cliente.cli_Nombre,      cli_Apellido,      Vehiculo.car_Marca,
Vehiculo.car_Modelo, Vehiculo.car_Costo
from (select car_Marca,car_Modelo,car_Costo, fk_Cliente
      from Carro,Venta
      where car_Codigo = fk_Carro) Vehiculo, Cliente
where cli_Cedula = Vehiculo.fk_Cliente and Vehiculo.car_Costo <= 50000;
```

**(Q11)**

```
Select  cli_Nombre,  cli_Apellido,  cli_Cedula,  cli_Telefono,  ven_Fecha,
ven_CostoTotal
FROM Cliente JOIN Venta
ON Cliente.cli_Cedula = Venta.fk_Cliente;
```

**(Q12)**

```
Select  cli_Nombre,  cli_Apellido,  cli_Cedula,  cli_Telefono,  ven_Fecha,
ven_CostoTotal
FROM Cliente INNER JOIN Venta
ON Cliente.cli_Cedula = Venta.fk_Cliente;
```

**(Q13)**

```
Select car_Marca, car_Modelo, ven_Fecha, ven_CostoTotal
FROM Carro RIGHT JOIN Venta
```

ON Carro.car\_Codigo = Venta.fk\_Carro;

**(Q14)**

Create index INDX\_CliEdad

ON Cliente (cli\_Apellido);

Select cli\_Nombre, cli\_Apellido

From cliente

where cli\_Apellido like '%z%' or cli\_Apellido like '%Z%';

**(Q15)**

Select ven\_Fecha, AVG(ven\_CostoTotal) as PromCT

FROM Venta

GROUP BY ven\_Fecha;

**(Q16)**

Select car\_Marca, SUM(car\_Costo) as SumaTotal, COUNT(car\_Marca) as  
Cantidad

FROM Carro

GROUP BY car\_Marca;

**(Q17)**

INSERT INTO CLIENTE VALUES ('1631112784499', 'Giacomo', 'Hammond',  
'7380170', 43);

**(Q18)**

UPDATE Carro

SET car\_Costo = car\_Costo\*1.12

WHERE car\_Marca = 'Audi';

**(Q19)**

UPDATE Carro

SET car\_Costo = car\_Costo\*1.05;

**(Q20)**

DELETE FROM Venta

WHERE ven\_Fecha <= '2000-01-01';

**(Q21)**

DELETE FROM Venta;

## 6.5 Bibliografía.

- © Microsoft. (2017). Información sobre las versiones de Windows 10: rama actual, historial de compilación.
- Almería, U. (n.d.). Tema 3. Gestión de memoria. Diseño de Sistemas Operativos.
- Andreas Schmitz. (2016). What is SAP HANA 2? | SAP News Center. Retrieved May 22, 2017, from <http://news.sap.com/what-is-sap-hana-2-nine-questions-answered/>
- Anguiano Morales, J. D. (2014). Características y tipos de bases de datos.
- Borghello, C. (2016). Database Benchmark: herramienta para stress test de bases de datos. ~ Segu-Info.
- Canto, C. (n.d.). Tipos de Memoria Tipos de Memoria Microprocesadores TIPOS DE MEMORIAS TIPOS DE MEMORIAS. Retrieved from [http://galia.fc.uaslp.mx/~cantocar/microprocesadores/EL\\_Z80\\_PDF\\_S/13\\_TIPOS\\_DE\\_MEMORIA.PDF](http://galia.fc.uaslp.mx/~cantocar/microprocesadores/EL_Z80_PDF_S/13_TIPOS_DE_MEMORIA.PDF)
- Culebro Juárez, M., Gómez Herrera, W., & Torres Sánchez, S. (2006). Software libre vs software propietario Ventajas y desventajas.
- Darmont, J. (n.d.). #275 Database Benchmarks.
- DB-Engines. (2017). DB-Engines Ranking - popularity ranking of database management systems.
- Diaconu, C., Freedman, C., Ismert, E., Larson, -Åke, Mittal, P., Stonecipher, R., ... Microsoft, M. Z. (n.d.). Hekaton: SQL Server's Memory-Optimized OLTP Engine. Retrieved from <https://www.microsoft.com/en-us/research/wp-content/uploads/2013/06/Hekaton-Sigmod2013-final.pdf>
- Elmasri, R., & Navathe, S. (2014). *Fundamentos de Sistemas de Bases de Datos. Porcentaje de Fe en vino blanco* (Vol. Primer). [https://doi.org/10: 0-8400-5444-0](https://doi.org/10.0-8400-5444-0)
- Espin, J. (2014). Innovacion en Oracle Database 12c - Joan Espin.
- Forrester Wave. (2017). The Forrester Wave™: In-Memory Databases, Q1 2017.
- González Barahona, J., Seoane Pascual, J., & Robles, G. (2003). *Introducción al software libre*.
- Inforges. (2013). SAP HANA, la computación en memoria - YouTube. Retrieved

- May 22, 2017, from <https://www.youtube.com/watch?v=6xv8tt6HLpU>
- Kay Somers. (2012). SAP HANA – Core Architecture. Retrieved May 22, 2017, from <http://en.community.dell.com/techcenter/b/techcenter/archive/2012/09/28/sap-hana-core-architecture>
- López, I. V., & Gutiérrez, G. B. (2009). El Benchmark TPC-H en MySQL y PostgreSQL. *Ingeniería.Lm.Uasnet.Mx*, 1–7.
- Marqués, M. (2009). *Bases de datos. Castellón de la Plana, ES: Universitat Jaume I. Servei de Comunicació i Publicacions.*
- McObject LLC. (2017). In-Memory Database Systems Questions and Answers.
- Microsoft. (n.d.). Documentación técnica de SQL Server 2016. Retrieved May 16, 2017, from <https://msdn.microsoft.com/es-es/library/ms130214.aspx>
- MSFT, J. T.-. (2014). New functionality in SQL Server 2014 – Part4 – In-Memory OLTP | Running SAP Applications on the Microsoft Platform. Retrieved May 16, 2017, from <https://blogs.msdn.microsoft.com/saponsqlserver/2014/02/09/new-functionality-in-sql-server-2014-part4-in-memory-oltp/>
- MySQL :: MySQL 5.7 Reference Manual :: 15.11 Overview of MySQL Storage Engine Architecture. (2017).
- MySQL ENTERPRISE EDITION. (n.d.).
- Oracle Data sheet | TimesTen In-Memory Database. (2014).
- Oracle White Paper | Extreme Performance Using Oracle TimesTen In-Memory Database. (2014).
- Patterson, D. A., & Hennessy, J. L. (2011). *Estructura y diseño de computadores.*
- Prakash Karia. (2013). In-memory OLTP Engine (Hekaton) in SQL Server 2014 CTP2. Retrieved May 16, 2017, from <https://www.pythian.com/blog/in-memory-oltp-engine-hekaton-in-sql-server-2014-ctp2/>
- Puertas Serrano, Á., & Delibes, M. (2007). Funcionamiento Interno Del Ordenador.
- RAIMA INC. (2017). In-Memory Database For Fast Applications | Raima.
- Red Gate Software Ltd. (2017). SQL Data Generator - Data Generator For MS SQL Server Databases.
- Romero, D. (2009). Gestión de memoria. Retrieved April 4, 2017, from

“Análisis benchmark de base de datos en memoria, base de datos en disco y base de datos mixta”

<http://puceftp.puce.edu.ec:2057/lib/pucesp/reader.action?docID=10316446>

Silberschatz, A. (Bell L., Korth, H. F. (Bell L., & Sudarshan, S. (Instituto Indio de Tecnología, B. (2002). *Fundamentos de bases de datos*.

<https://doi.org/10.1017/CBO9781107415324.004>

Silberschatz, A., Galvin, P., & Gagne, G. (2006). *Fundamentos de sistemas operativos* (Séptima).

SQL Server 2016 | Microsoft. (2017).

Stallman, R. M. (2004). Software libre para una sociedad libre.

Tellez, A. F. (n.d.). CAPÍTULO 1 CONCEPTO DE COMPUTADOR. Retrieved from <http://robotica.uv.es/pub/Libro/PDFs/CAP11.pdf>

The PostgreSQL Global Development Group. (n.d.). Sobre PostgreSQL | [www.postgresql.org.es](http://www.postgresql.org.es). Retrieved May 19, 2017, from <https://www.postgresql.org/about/>

The PostgreSQL Global Development Group. (2017). PostgreSQL: Documentation: 9.6: PostgreSQL 9.6.3 Documentation. Retrieved May 19, 2017, from <https://www.postgresql.org/docs/9.6/static/index.html>

TPC-Homepage V5. (2017).

What is a Hybrid Database? (n.d.).

Zaunere, H. (2006). MySQL-RDBMS-Technical-Overview.